Multilevel Transitive and Intransitive Non-Interference, Causally[☆]

Paolo Baldan[a,], Alessandro Beggiato[b]

[a]*Dipartimento di Matematica, Università di Padova, Italia*
[b]*IMT School for Advanced Studies Lucca, Italia*

Abstract

We develop a theory of non-interference for multilevel security based on causality, with Petri nets as a reference model. We first focus on transitive non-interference, where the relation representing the admitted flow is transitive. Then we extend the approach to intransitive non-interference, where the transitivity assumption is dismissed, leading to a framework which is suited to model a controlled disclosure of information. Efficient verification algorithms based on the unfolding semantics of Petri nets stem out of the theory. We also argue about the possibility of performing a compositional verification.

Keywords: multilevel non-interference, intransitive policies and downgrading, Petri nets, unfolding semantics, true concurrency, verification

1. Introduction

The problem of controlling the flow of information in a computing system is a classical one, faced by many contributions in the literature. A general formalization of information flow security is provided by [1] that introduces the notion of non-interference. Intuitively a security level is said not to interfere with another if what can be observed at the latter level is not affected by what happens at the former. In the simplest scenario, entities are classified according only to two levels, a *High* level, which intuitively should be confidential, and a *Low* level, which is public, and the information is allowed to flow from *Low* to *High*, but not vice-versa.

Different notions of behavior and observation lead to different non-interference properties. Originally non-interference has been studied for deterministic sequential systems, relying on a trace semantics. Since then, several variants of non-interference have been studied, dealing with concurrent and non-deterministic systems. In concurrent formalisms which offer forms of composition and synchronization, such as process calculi

[☆]Work supported by the MIUR PRIN project CINA and by the University of Padova project ANCORE.

[*]Corresponding author
Email addresses: baldan@math.unipd.it (Paolo Baldan), alessandro.beggiato@imtlucca.it (Alessandro Beggiato)

and Petri nets, a popular formulation of non-interference is the so-called NDC (Non-Deducibility on Composition), which looks at the system under analysis as a component, possibly interacting with the surrounding environment. It states that a process (or net) S is free of interferences whenever S running in isolation, seen from the low level, is behaviorally equivalent to S interacting with any parallel high level process (or net) that may synchronize on high actions (or transitions) [2, 3, 4, 5, 6, 7, 8, 9]. Intuitively, this is often described by referring to some informal notion of causality – the activity at high level should not cause any visible effects on the behavior at low level – but formalized in terms of interleaving semantics.

This informal reference to causality is made formal in [7] that, relying on some previous work on non-interference notions in contact-free elementary nets (or equivalently pure safe nets) and trace nets [5], provides a causal characterization of BNDC (Bisimulation-based NDC) for Petri nets, in terms of the unfolding semantics [10]. The interest for a causal characterization is not only of theoretical nature. On the pragmatic side the use of a true concurrent semantics, like the unfolding, which represents interleaving only implicitly, is helpful to face the state explosion problem which affects the verification of concurrent systems.

Since its infancy (see, e.g., [11]) information flow security has recognized the usefulness of dealing with multilevel security domains, where the security levels are not limited to "high" and "low". In general, a domain of security levels is considered, with a relation between levels specifying the admitted flows. The transitive nature of information flow – if information flows from level A to level B and from B to C then it necessarily flows from A to C – naturally leads to work with security domains where the admitted flow relation is a partial order and a system policy of the kind no read-up, no write-down, only allowing a flow of information from lower to higher levels. The order can be total, expressing a hierarchy of confidentiality degrees (e.g., top secret, secret, confidential and unclassified in a military setting). It can also be partial, typically when various confidentiality criteria are combined into a single domain. For instance, an administration could keep public and sensitive citizen data concerning taxes and civil status. The fact that the rights of accessing sensitive tax and civil status data are independent, naturally leads to a lattice of security levels.

As argued, e.g., in [12] it can also be natural to consider security policies where the admitted flow relation is not transitive, in a way that a direct flow between two security levels, say from A to B, is forbidden, while a flow mediated through a third level, say D, is admitted. Intransitive policies are suited, for instance, for representing declassification or downgrading of confidential information. This allows for a controlled form of leakage, making such policies more realistic than pure non-interference policies that instead impose a complete isolation of confidential levels. More generally, by exploiting intransitive policies, it is possible to prescribe the (possibly cyclic) paths on which information is allowed to flow in a given system.

In this paper, building on [7, 13], we provide a causal characterization of non-interference properties for (safe) Petri nets in a multilevel setting. We first focus on multilevel transitive policies and the property BNDC. Then we consider intransitive policies and the property BINI (Bisimilarity-based Intransitive Non-Interference) the adaptation of BNDC to intransitive security domains. The characterization is used to develop corresponding verification algorithms based on the unfolding semantics, that are implemented in a tool called MultiUBIC (Multi Unfolding-Based Interference Checker).

More in detail, in the transitive case a Petri net is shown to enjoy BNDC when its unfolding reveals neither a direct causality from a higher level transition to a lower level one (witnessed by a *weak causal* place), nor a direct conflict between a lower level and a higher level transition (witnessed by a *weak conflict* place). Both situations represent a violation of the policy: in the first situation, intuitively, a token produced at a higher level flows down to a lower level, while in the second situation a transition of a higher level competes for a token with one at a lower level. In the intransitive case, the characterization becomes slightly more complex: a violation of the policy is still witnessed by an influence (causality or conflict) from some level A to a level B to which the flow is not permitted, but this must not be mediated by a level where information can legitimately flow from A. Such characterizations enable the definition of algorithms that check the non-interference property on suitably defined complete prefixes of the unfolding.

Relying on the causal characterization, we also prove some compositionality properties of transitive and intransitive non-interference, that can be of help in reducing the complexity of the verification phase. In particular we show that, when a system can be decomposed as the parallel composition of subcomponents, the absence of interferences (validity of BNDC or BINI) for the entire system can be deduced from the absence of interferences in the subcomponents.

The unfolding-based algorithms are implemented in the tool MultiUBIC [14]. Compared to tools that construct (or explore) the reachability graph of the net, like ANICA (Automated Non-Interference Check Assistant) [15] and PNSC (Petri Net Security Checker) [16], the partial order representation of concurrency in MultiUBIC - as for its predecessor UBIC – leads to a gain of efficiency for highly concurrent systems where the unfolding prefix can be exponentially smaller than the complete state space (see e.g. [17]).

In the paper we also show that the verification of multilevel policies can be reduced to a number of problems on two-level security domains (possibly enriched with a downgrading level in the intransitive case). This suggests an alternative way of dealing with multilevel systems. Indeed, MultiUBIC comes equipped with facilities for performing the reduction. The experiments suggest that, in general, a direct multilevel verification is more efficient when the number of levels increases, but situations are singled out where the reduction is instead more convenient.

This paper brings to a maturity the work initiated in [7, 13]. Concerning the transitive setting, we generalize [7] by developing notions, algorithms and a tool that deal with general multilevel domains rather than with two-level domains. Once the right notions are identified some parts of the extension work relatively smoothly. Hence we tried to describe only the main aspects, still keeping the paper as much as possible self-contained. Concerning the intransitive case, the paper treats general multilevel intransitive domains that include, as a special case, the two-level domains with downgrading of the conference paper [13].

The rest of the paper is organized as follows. In § 2 we define multilevel security domains and we review some basic notions for Petri nets, and their unfolding semantics. In § 3 we focus on transitive policies and the BNDC property, providing a causal characterization and a corresponding algorithm for verifying whether a safe Petri net is BNDC. In § 4 we extend the results established for BNDC to intransitive policies. In § 5 we prove some compositionality properties for BNDC and BINI. In § 6 we present the tool MultiUBIC and discuss the results of some test runs (fully detailed in Appendix B). In § 7 we draw some conclusions and outline possible research directions. Results and

algorithms presented for safe nets can be easily extended to the so-called locally-safe nets, a larger class of possibly non-safe Petri nets defined in [13]. For the sake of simplicity, in the main text we deal with safe nets. The (small) changes needed to adapt the theory to locally-safe nets are described in Appendix A.

2. Multilevel Security Domains and Petri Nets

In this section, we introduce multilevel security domains and we review some basic notions about Petri nets, with special attention to their unfolding semantic. xs

2.1. Multilevel Security Domains

We start by defining the notion of security domain.

Definition 1 (multilevel security domain). *A* multilevel security domain $(\mathscr{L}, \rightsquigarrow)$ *is a finite set of security levels $\mathscr{L}$, endowed with a binary reflexive relation $\rightsquigarrow \subseteq \mathscr{L} \times \mathscr{L}$ called* security policy*. If $\rightsquigarrow$ is also transitive and thus a preorder we call $(\mathscr{L}, \rightsquigarrow)$ a* transitive multilevel security domain*.*

The security policy describes which information flows are legitimate. It is assumed to be reflexive because entities at the same security level should reasonably be able to freely exchange information. Concerning transitivity, as already observed, while the flow of information is transitive by its nature, security policies need not to be. Security levels will be ranged over by L, L', L_1, L'_1 and so forth. Without loss of generality we can assume any transitive security domain to be a partial order, i.e., the relation $\rightsquigarrow$ to be antisymmetric. In fact, if $\rightsquigarrow$ is a proper preorder (i.e, it is not antisymmetric), we can equivalently consider the induced partial order obtained as the quotient under the equivalence $\rightsquigarrow \cap \rightsquigarrow^{-1}$. In fact, since equivalent levels can communicate in either direction, they can be safely collapsed. Examples of multilevel security domains will be discussed later, after introducing also net systems. Given $S \subseteq \mathscr{L}$ we write $\overline{S}$ for its complement $\mathscr{L} \setminus S$.

Definition 2 (upper sets and targets). *Let $(\mathscr{L}, \rightsquigarrow)$ be a multilevel security domain. An* upper set *is a subset $U \subseteq \mathscr{L}$ such that if $L \in U$ and $L \rightsquigarrow L'$ then $L' \in U$. The set of* targets *of a security level $L \in \mathscr{L}$ is $\uparrow L = \{L' \in \mathscr{L} \mid L \rightsquigarrow L'\}$. The set of* strict targets *of L is $\Uparrow L = \uparrow L \setminus \{L\}$.*

Intuitively, an entity (user, program, variable, instruction) with associated security level L has permission to influence, or to write, or to pass information to any entity with security level in $\uparrow L$. Any other information flow is a violation of the policy. Notice that a violation involves two levels: it is a flow from level L to level L' with $L \not\rightsquigarrow L'$. It can be perceived from both ends with different perspectives: L is "talking" to so someone it should not, or L' is "listening" something it should not.

2.2. Petri Nets and Net Systems

A *(Petri) net* is a tuple $N = (P, T, F)$ where P, T are two disjoint sets of *places* and *transitions*, respectively, and $F : (P \times T) \cup (T \times P) \to \mathbb{N}$ is the *flow function*. Graphically places and transitions are drawn as circles and rectangles, respectively, while the flow function is represented by using weighted directed arcs connecting places and transitions. For example if $F(p, t) = 2$ then there is an arc from p to t of weight 2. When the arc weight is 1 it will be omitted. For $x \in P \cup T$ we define ${}^\bullet x = \{y \in P \cup T : F(y, x) > 0\}$ (the *pre-set* of x) and $x^\bullet = \{y \in P \cup T : F(x, y) > 0\}$ (the *post-set* of x). We will use ${}^\bullet(.)$ and $(.)^\bullet$ also over sets, letting ${}^\bullet X = \bigcup_{x \in X} {}^\bullet x$ and $X^\bullet = \bigcup_{x \in X} x^\bullet$, for any $X \subseteq P \cup T$. For all nets in the paper we will assume that ${}^\bullet t \neq \emptyset$ for all $t \in T$.

A *marking* of N is a function $m : P \to \mathbb{N}$. A transition $t \in T$ is *enabled* at a marking m, denoted $m[t\rangle$, if $m(p) \geq F(p, t)$ for all $p \in P$. If $m[t\rangle$ then t can be *fired* leading to a new marking m', written $m[t\rangle m'$, defined by $m'(p) = m(p) + F(t, p) - F(p, t)$ for all places $p \in P$. The enabling and firing relations are extended to $\sigma \in T^*$ (set of all finite sequences of elements of T) by defining $m[\varepsilon\rangle m$ (where ε is the empty sequence) and $m[\sigma\rangle m'[t\rangle m''$ imply $m[\sigma t\rangle m''$. For a marking m, we denote $m^\circ = \{t \in T : m[t\rangle\}$. In pictures markings are represented as black dots, called *tokens*, inside places (the presence of n dots inside place p means that $m(p) = n$). A *marked net* is a pair $\mathbf{N} = (N, m_0)$ where N is a net and m_0 is a marking of N. Since marked nets can be recognized by the use of the boldface symbol, the qualification "marked" will be sometimes omitted. A marking m' is *reachable* if there exists $\sigma \in T^*$ such that $m_0[\sigma\rangle m'$. The set of reachable markings of $\mathbf{N}$ is denoted by $[m_0\rangle$. When $m[t\rangle m'$, the marking m', uniquely determined by m and t, is denoted by $\langle m[t\rangle$. Analogously, for $\sigma \in T^*$, if $m[\sigma\rangle$ we can define the marking $\langle m[\sigma\rangle$.

A net $\mathbf{N}$ is *safe* if for every $p \in P$ and every $m \in [m_0\rangle$ we have $m(p) \leq 1$.

In order to formalize information flow properties in the setting of Petri nets, as in [5, 6], we work with Petri nets where transitions are associated with security levels taken from a fixed multilevel domain. Differently from [5, 6] levels are not confined to be just *High* and *Low*.

Definition 3 (net system). *Given a multilevel security domain $\mathscr{L}$, a* net system *over $\mathscr{L}$ is a tuple $N = (P, T, F, \lambda)$ where (P, T, F) is a Petri net and $\lambda : T \to \mathscr{L}$ is a function which assigns a security level to each transition. For $S \subseteq \mathscr{L}$ we define $T_S = \{t \in T \mid \lambda(t) \in S\}$, the set of transitions whose security level is in S.*

Net systems will typically be ranged over by N, N', N_0 and so on. Superscripts and postscripts carry over the components of the net system. With a slight abuse of notation, we will write T_L instead of $T_{\{L\}}$. For the sake of conciseness, we will omit the parentheses when applying λ to a transition, writing λt for $\lambda(t)$. Moreover, we will apply λ to sets of transitions $T' \subseteq T$, writing $\lambda T'$ for the set $\{\lambda t \mid t \in T'\}$.

As an example, consider the net system $\mathbf{S}$ and security domain in Fig. 1. It represents a measurement device consisting of two independent sensors that get new data for a processor, that, in turn, can poll them to acquire a more recent measurement. Each sensor has a cyclic behavior. For instance, the left sensor is capable to get a measure (transition get_A). Such measure can be exposed at its interface (transition $show_A$) and then removed after a while (transition rem_A), restarting the cycle. Alternatively, the measure can be sent to a shared cache (transition $send_A$) which in turn update the

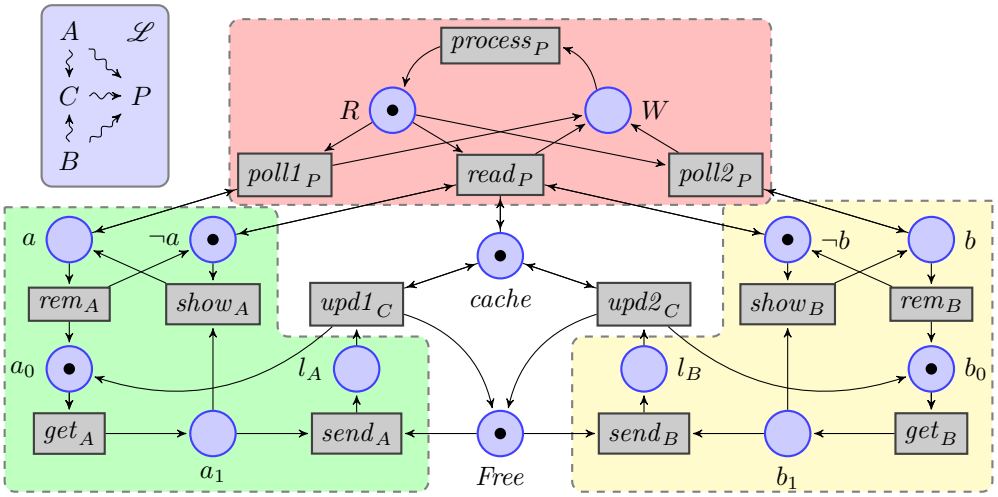


Figure 1: A non-BNDC net system $\mathbf{S}$ over the security domain $\mathscr{L}$ (top left). Function λ is $\lambda t_L = L$.

memory (transition upd_C). Note that the presence or absence of a datum at the interface is represented by a token in place a or $\neg a$, respectively. The cache is accessed by the two sensors via transitions $upd\,i_C$ in mutual exclusion (since it can store a single measure), as guaranteed by the use of place *Free*, consumed by transitions $send_A$ and $send_B$, and produced by $upd1_C$ and $upd2_C$. The processor cyclically get some value for the measure. If a value is exposed at the sensor interfaces (places a or b marked) then one of such values is taken (transitions $poll1_P$ and $poll2_P$), otherwise (places $\neg a$ and $\neg b$ marked) the value from the cache is read (transition $read_P$).

The security level of transitions is given by their subscript (namely, $\lambda t_L \mapsto L$). Transitions modeling the left and right sensors have security level A and B, respectively. The cache and processor have security levels C and P respectively. The intuition is that the two sensors should not interfere with each other, and they can send information to the processor directly or through the cache. The processor and the cache should not affect the behavior of the sensors. Throughout the paper we will consistently adopt the same graphical notation of Fig. 1 for net systems: transitions will be annotated with their level as a subscript, and the associated domain will be drawn in a (blue) box.

An *S-system* is a net system such that $T = T_S$, i.e. a system only capable of performing actions with security level in S. We call $\mathbf{N} = (N, m_0)$ a marked S-system when N is an S-system, and since we will consistently use the bold font to denote marked systems we will often drop the qualification "marked".

In order to formalize the non-interference notions we resort, as in [6], to some operations on nets systems, namely (parallel) composition and restriction.

Definition 4 (composition). *Let N and N' be two net systems such that $P \cap P' = \emptyset$ and for all $t \in T \cap T'$ it holds $\lambda t = \lambda' t$. The* composition *of N and N' is the net system $N \,|\, N' = (P \cup P', T \cup T, \lambda \cup \lambda', F \cup F')$. The composition of $\mathbf{N} = (N, m_0)$ and $\mathbf{N}' = (N', m'_0)$ is the marked net system $\mathbf{N} \,|\, \mathbf{N}' = (N \,|\, N', m_0 \cup m'_0)$.*

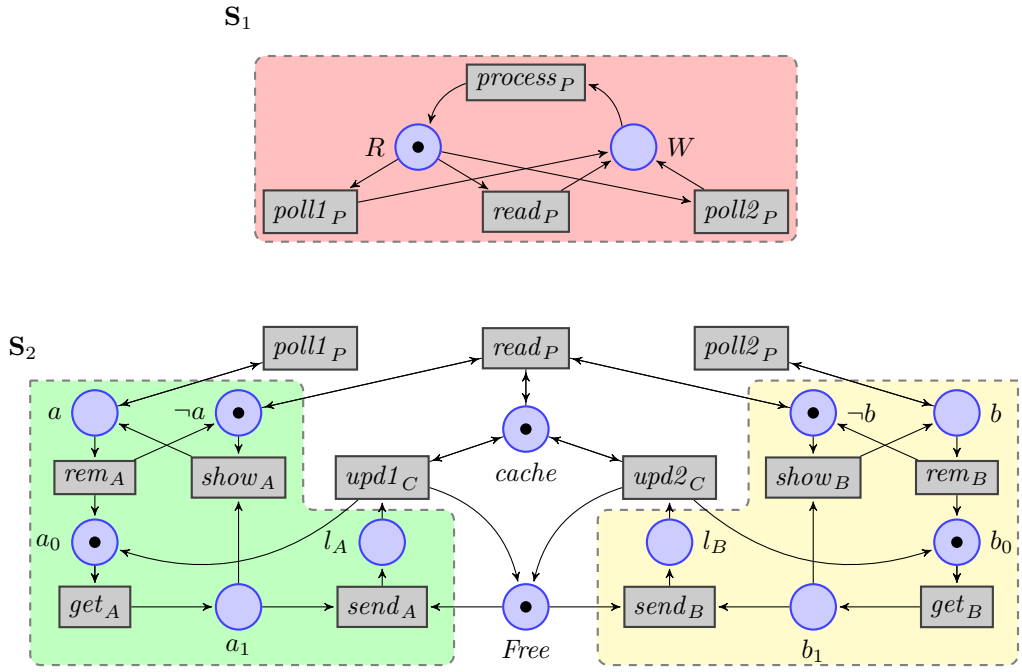


Figure 2: Two subsystems $\mathbf{S}_1$ and $\mathbf{S}_2$ of the running example $\mathbf{S}$ such that $\mathbf{S} = \mathbf{S}_1 \,|\, \mathbf{S}_2$.

Note that above $F \cup F'$ and $m_0 \cup m_0'$ are well-defined because of the disjointness conditions on the set of places of N and N', and $\lambda \cup \lambda'$ is well defined because we required the labeling functions to agree over the common transitions of N and N'.

Intuitively $N \,|\, N'$ is the parallel composition of N and N' synchronized on the common transitions. Whenever we consider two net systems N and N' we shall implicitly assume that the disjointness requirements are satisfied so that $N \,|\, N'$ always makes sense. Note that this might require some "renaming".

As an example, the net system $\mathbf{S}$ of Fig. 1 can be obtained as the composition $\mathbf{S}_1 \,|\, \mathbf{S}_2$ where $\mathbf{S}_1$ and $\mathbf{S}_2$ are two subsystem reported in Fig. 2.

The restriction of net system with respect to a set of transitions T' simply removes the transitions in T'. The formal definition follows.

Definition 5 (restriction). *Given a net system N and a subset $T_1 \subseteq T$, the* restriction *of N by T_1 is the net system $N \backslash T_1 = (P, T \backslash T_1, \lambda \backslash T_1, F \backslash T_1)$ where $\lambda \backslash T_1$ is the restriction of λ to T_1 and $F \setminus T_1$ is the restriction of F to $(P \times (T \setminus T_1)) \cup ((T \setminus T_1) \times P)$. For a net system $\mathbf{N}$, the restriction $\mathbf{N} \setminus T_1$ is $(N \setminus T_1, m_0)$.*

As an example, the restriction $\mathbf{S} \setminus T'$, where $\mathbf{S}$ is the running example and $T' = \{rem_A, show_A, get_A, send_A\}$ can be found in Fig. 3.

2.3. Unfolding of Net Systems

The unfolding of a Petri net $\mathbf{N}$ is a structure that provides a compact representation of the possible computations of $\mathbf{N}$: places represent occurrences of the tokens that

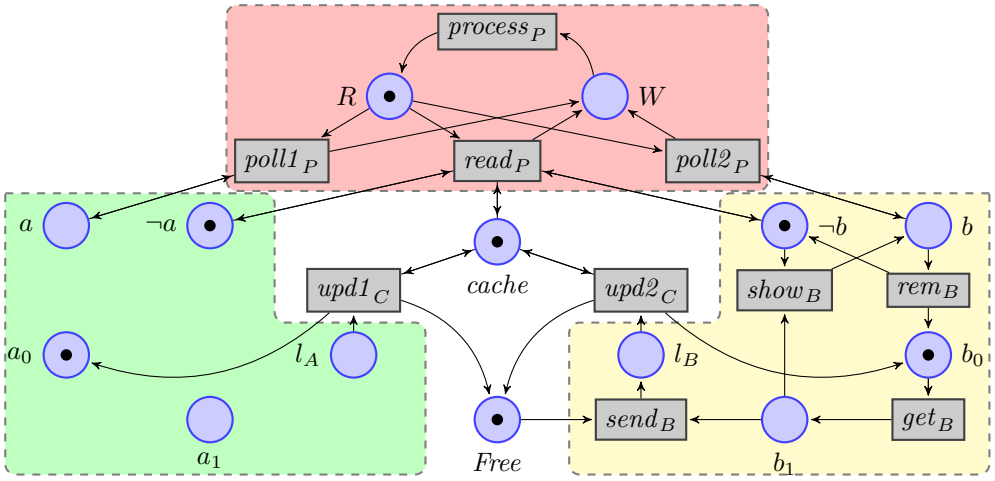


Figure 3: The restriction $\mathbf{S} \setminus T'$ with $T' = \{rem_A, show_A, get_A, send_A\}$.

are produced in computations of $\mathbf{N}$ and transitions are copies of the transitions of $\mathbf{N}$, representing their possible firings [10]. The unfolding is infinite whenever the net $\mathbf{N}$ has a cyclic behavior, but when the net is finite state, along the lines of the seminal work in [18, 19], finite fragments can be constructed containing a representation of all the reachable markings. For highly concurrent systems, such prefixes are possibly exponentially smaller than the so-called marking graph, i.e., the directed graph whose nodes are the reachable markings of $\mathbf{N}$ and arcs correspond to transition firings.

We next define some relevant dependency relations that, on net unfoldings, allow us to characterize computations and reachable markings.

Definition 6 (dependency relations). *Let N be a net. The* causality *relation $<$ is the least transitive binary relation on $P \cup T$ such that $x < y$ if $x \in {}^\bullet y$. By $\leq$ we denote the reflexive closure of $<$. The* conflict *relation $\sharp$ is the least symmetric binary relation on $P \cup T$ such that (i) if $t, t' \in T$, $t \neq t'$ and ${}^\bullet t \cap {}^\bullet t' \neq \emptyset$ then $t \sharp t'$; (ii) if $x < x'$ and $x \sharp y$ then $x' \sharp y$. We say that x, x' are* concurrent *when neither $x < x'$ nor $x' < x$ nor $x \# x'$.*

The unfolding is an acyclic net, constructed inductively starting from the initial marking of $\mathbf{N}$ and then adding, at each step, an occurrence of each transition of $\mathbf{N}$ which is enabled by (the image of) a concurrent subset of the places already generated. Below, we write π_1 for the standard projection function over the first component of pairs.

Definition 7 (unfolding). *Let $\mathbf{N} = ((P, T, F), m_0)$ be a marked net. Define the net $U^{(0)} = (T^{(0)}, P^{(0)}, F^{(0)})$ as follows:*

$$T^{(0)} = \emptyset \ ; \qquad P^{(0)} = \bigcup_{p \in P} \{(p, i) : 1 \leq i \leq m_0(p)\} \ ; \qquad F^{(0)} = \emptyset \ .$$

Then we define the unfolding as the least net $\mathcal{U}(\mathbf{N}) = (P^{(\omega)}, T^{(\omega)}, F^{(\omega)})$ containing $U^{(0)}$ and such that

- *if* $t \in T$ *and* $X \subseteq P^{(\omega)}$ *is a pairwise concurrent set of places such that* $\pi_1(X) = {}^\bullet t$, *and* $\forall p \in {}^\bullet t.\ |\{b \in X : \pi_1(b) = p\}| = F(p,t)$, *then* $y = (t, X) \in T^{(\omega)}$; *moreover* $F^{(\omega)}(x, y) = 1$ *for all* $x \in X$.
- *if* $t \in T^{(\omega)}$, $p \in \pi_1(t)^\bullet$ *and* $1 \leq i \leq F(\pi_1(t), p)$, *then* $z_i = (p, t, i) \in P^{(\omega)}$; *moreover* $F^{(\omega)}(t, z_i) = 1$ *for all* z_i *(*$i = 1, \ldots, F(\pi_1(t), p)$*).*

The unfolding of a marked net system falls into the class of *occurrence nets* [10], a subclass of nets where causality is acyclic and well-founded, conflict is irreflexive and the arcs of the flow relation have weight at most 1 (i.e., F is a relation). The initial marking is often left implicit as it is identified as the set of minimal places.

Places and transitions in the unfolding represent tokens and firing of transitions, respectively, of the original net. Each place in the unfolding is a tuple recording the place in the original net and the "history" of the token. For historical reasons transitions and places in the unfolding are also called *events* and *conditions*, respectively. The projection π_1 over the first component maps places and transitions of the unfolding to the corresponding items of the original net $\mathbf{N}$.

Notation 1. *Given an event* e, *the security level of the corresponding transition* $\pi_1(e)$ *will be often referred to as the security level of the event and we will write* λe *for* $\lambda\pi_1(e)$.

As an example, consider the net system $\mathbf{N}$ in Fig. 4 (left). This is a slightly simplified version of the subnet that, in the running example of Fig. 1, corresponds to one of the sensors. A fragment of the unfolding $\mathcal{U}(\mathbf{N})$ of such system is provided in Fig. 4(right). In the unfolding, the conditions labeled by a_0 and $\neg a$ on the top, according to Definition 7, are $(a_0, 1)$ and $(\neg a, 1)$, respectively. Event get^1_A is $(get_A, \{(a_0, 1)\})$ and the condition a_1 in its post-set is $(a_1, get^1_A, 1)$. Similarly, event $show^1_A$ is $(show_A, \{(a_1, get^1_A, 1), (\neg a, 1)\})$. As examples of dependency relations, note that $get^1_A \leq show^1_A$ and $get^1_A \leq send^1_A$, while $show^1_A \sharp send^1_A$ and $show^1_A \sharp upd^C_1$. In this specific case there are no events which are neither causally dependent nor in conflict, hence no pair of them is concurrent. Concurrency will come into play later, when dealing with the full system where we have two copies of the sensor.

The runs of $\mathbf{N}$ are represented by the configurations of $\mathcal{U}(\mathbf{N})$, i.e., subsets of $T^{(\omega)}$ that are causally closed and conflict-free. For an event $e \in T^{(\omega)}$ we define its *causes* as the set $[e] = \{e' \in T^{(\omega)} : e' \leq e\}$. We write $[e)$ for the set of strict causes, i.e., $[e) = [e] \setminus \{e\}$. We extend the notion of set of causes to subsets $X \subseteq T^{(\omega)}$ by setting $[X] = \bigcup_{e \in X} [e]$.

Definition 8 (configuration). *A* configuration *of* $\mathcal{U}(\mathbf{N})$ *is a finite subset* $C \subseteq T^{(\omega)}$ *such that* $(C \times C) \cap \sharp = \emptyset$ *and* $[C] = C$. *The set of all configurations of* $\mathcal{U}(\mathbf{N})$ *is denoted by* $\mathcal{C}(\mathcal{U}(\mathbf{N}))$.

A configuration of $\mathcal{U}(\mathbf{N})$ can be associated with a reachable marking of $\mathbf{N}$, obtained by firing all its events in any order compatible with causality. Formally, we define the cut of a configuration C as the set of places $C^\circ = (P^{(0)} \cup \bigcup_{e \in C} e^\bullet) \setminus (\bigcup_{e \in C} {}^\bullet e)$, which is the marking of the unfolding reached after the execution of C. This in turn induces a marking on $\mathbf{N}$ given by $\mathsf{M}(C)(p) = |\{b \in C^\circ : \pi_1(b) = p\}|$, for every place p of $\mathbf{N}$.

The unfolding can be shown to be *marking complete* in the sense that $m \in [m_0\rangle$ if and only if there exists $C \in \mathcal{C}(\mathcal{U}(\mathbf{N}))$ such that $\mathsf{M}(C) = m$ (see [10, 20, 18]).

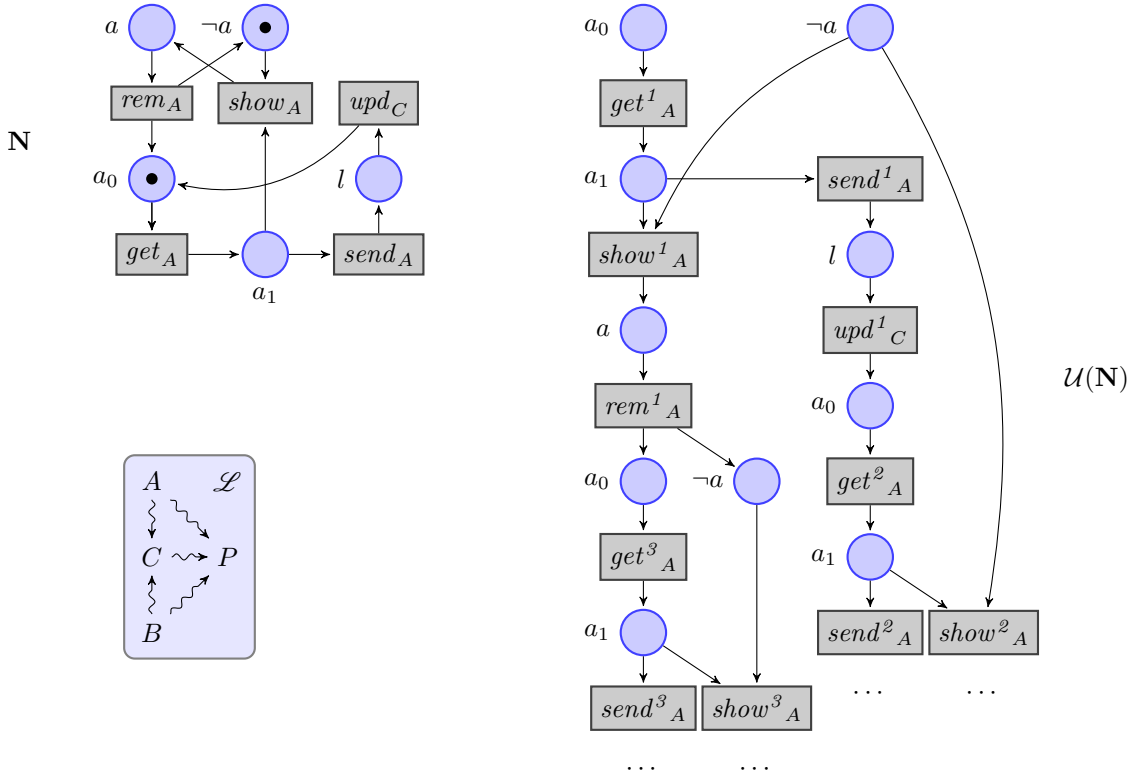


Figure 4: A net system and the initial part of its unfolding.

3. Transitive Multilevel Non-Interference

In this section we focus on transitive multilevel security domains and we define the reference security property in the paper as an instance of (Bisimulation-based) Non-Deducibility on Composition (BNDC).

3.1. Bisimilarity-based Non-Deducibility on Composition

Let $(\mathscr{L}, \rightsquigarrow)$ be a fixed transitive multilevel security domain, that throughout the section will be designated simply as $\mathscr{L}$. The definition of BNDC is obtained by adapting that in [5, 7] to the multilevel setting. First, in order to formalize the intuitive idea of variations of the behavior which are visible at a given security level we rely on a *view function* which filters any firing sequence by keeping only the transitions whose security level is in a given set S. This is sometimes called a *purge function* (see e.g. [21]).

Definition 9 (view function). *Given a subset of the domain $S \subseteq \mathscr{L}$ and any net*

system N, the view function $S(\bullet) : T^* \to T_S^*$, *is defined inductively by*

$$S(\sigma) = \begin{cases} \epsilon & \text{if } \sigma = \epsilon \\ tS(\sigma') & \text{if } \sigma = t\sigma' \text{ and } \lambda t \in S \\ S(\sigma') & \text{if } \sigma = t\sigma' \text{ and } \lambda t \notin S \end{cases}$$

The view function is used to define a bisimulation equivalence intended to capture the discriminating power of a user which is able to observe only events with security level in a given set.

Definition 10 (S-view bisimulation). *Let $\mathbf{N}$ and $\mathbf{N}'$ be net systems and $S \subseteq \mathscr{L}$. An* S-view simulation *of $\mathbf{N}$ by $\mathbf{N}'$ is a relation $R \subseteq [m_0\rangle \times [m_0'\rangle$ such that:*

- $(m_0, m_0') \in R$;
- *if $(m, m') \in R$ and $m[\sigma\rangle$ then there exists σ' such that $S(\sigma) = S(\sigma')$, $m'[\sigma'\rangle$ and $(\langle m[\sigma\rangle, \langle m'[\sigma'\rangle) \in R$.*

An S-view bisimulation *between $\mathbf{N}$ and $\mathbf{N}'$ is a relation $R \subseteq [m_0\rangle \times [m_0'\rangle$ such that both R and R^{-1} are S-view simulations. If there exists an S-view bisimulation between $\mathbf{N}$ and $\mathbf{N}'$, we say that they are* S-view bisimilar *and we write $\mathbf{N} \approx_S \mathbf{N}'$.*

When working in a two-level setting, namely in the domain $\mathscr{B} = \{Low \rightsquigarrow High\}$, a system is considered free of interferences when the low level behavior is not influenced by high level interactions. Formally, following [6], a net system $\mathbf{N}$ is BNDC when for any *High*-net system $\mathbf{N}'$ (where all transitions are labeled *High*),

$$\mathbf{N} \quad \approx_{Low} \quad (\mathbf{N} \,|\, \mathbf{N}') \setminus (T_{High} \setminus T')$$

i.e., the "low level" view of the behavior of $\mathbf{N}$ remains unchanged when the net interacts with any high level net system.

The generalization to the multilevel setting considers any partition of the security domain in an upper set $U \subseteq \mathscr{L}$ and its complement $\overline{U}$, and requires that U does not influence the view of $\overline{U}$.

Definition 11 (BNDC). *Let $\mathbf{N}$ be a net system. Given an upper set $U \subseteq \mathscr{L}$, we say that $\mathbf{N}$ is U-BNDC if for every marked U-system $\mathbf{N}'$ we have that*

$$\mathbf{N} \quad \approx_{\overline{U}} \quad (\mathbf{N} \,|\, \mathbf{N}') \setminus (T_U \setminus T').$$

The system is BNDC if it is U-BNDC for any upper set $U \subseteq \mathscr{L}$.

The definition can be understood as follows. Observe that, given an upper set U, if the system is not U-BNDC then there is a flow from some level $L \in U$ to some level $L' \in \overline{U}$. This flow is a security violation since $L \not\rightsquigarrow L'$ otherwise L' would be in U. Vice versa, if there is a security violation, it will consist of a flow of information from some security level L to a level L' which cannot be influenced by L, namely $L \not\rightsquigarrow L'$. This is captured by the definition above when considering the upper set $U = \uparrow L$. In fact, by hypothesis $L' \in \overline{U}$.

Clearly, in the two-level domain $\mathscr{B} = \{Low \rightsquigarrow High\}$, Definition 11 is the usual one. In fact, the only non-trivial partition is induced by the upper set $U = \{High\}$.

Note that the validity of the BNDC property for a multilevel system is reduced to the validity of BNDC in a number of two-level domains, one for each upper set, with U playing the role of the high part of the system, and its complement playing the role of the low part.

The considerations above suggests that, whenever a security violation exists, it can be detected when analyzing an upper set in the security domain of the kind $U = \uparrow L$ for some level L. This is indeed the case and we will prove it by exploiting the characterization of BNDC based on causal and conflict places in the next section (see Corollary 1).

3.2. Characterizing Multilevel BNDC through Causal and Conflict Places

In this section we provide a characterization of the BNDC property on multilevel domains based on causal and conflict places. Roughly speaking, a net system is shown to be BNDC when there is no causal flow which is not allowed by the security policy, i.e., when there is no causal dependency from a level L_1 to a level L_2 such that $L_1 \not\rightsquigarrow L_2$. Additionally, transitions with different security levels must never be in conflict, competing for a token. In fact, if transitions of different levels L_1 and L_2 were in conflict, each one would influence the behaviour of the other, and thus, since by antisymmetry, either $L_1 \not\rightsquigarrow L_2$ or $L_2 \not\rightsquigarrow L_1$, at least in one direction the influence would violate the policy. This generalizes the work developed for the two-level case in [5, 7].

Definition 12 (causal place). *Let H, $L \in \mathscr{L}$ be two security levels such that $H \not\rightsquigarrow L$. A place $p \in P$ of a net system $\mathbf{N}$ is HL-*causal *if:*

(i) $p \in {}^\bullet l \cap h^\bullet$ for transitions $h, l \in T$ such that $\lambda h = H$ and $\lambda l = L$;

(ii) there exists $m \in [m_0\rangle$ satisfying $m[h\tau l\rangle$, $m[\tau\rangle$ and $\langle m[\tau\rangle(p) < F(p,l)$, with $\tau \in T^*_{\overline{\uparrow H}}$.

Place p is called causal *if it is HL-causal for some levels $H, L \in \mathscr{L}$.*

By condition (i), transition l potentially consumes a token produced by transition h, despite the fact that a flow between their security levels is forbidden by the policy. By condition (ii) there is a firing sequence $h\tau l$ where this indeed happens. In fact, since l is enabled by $m[h\tau\rangle$ but not by $m[\tau\rangle$ with $\langle m[\tau\rangle(p) < F(p,l)$, it means that the firing of h generates a token in p which is essential for the firing of l. The requirement that $\tau \in T^*_{\overline{\uparrow H}}$ could be relaxed, allowing $\tau \in T^*$, without altering the theory. Intuitively, if the condition in Definition 12 is satisfied with a firing sequence $\tau \in T^*$ that includes some transition t with security level $H' \in \uparrow H$, say $\tau = \tau_1 t \tau_2$, then either t can be removed or we would have a violation from t to l, with τ_2 shorter than τ. In fact, note that since $H \rightsquigarrow H'$ and $H \not\rightsquigarrow L$, by transitivity we deduce $H' \not\rightsquigarrow L$. Since τ can be shortened a finite number of times, eventually we would end up in a sequence $\tau \in T^*_{\overline{\uparrow H}}$.

A causal place catches the presence of a positive information flow forbidden by the policy: observing l it is possible to deduce that h happened in the past, even not having the security level required to observe h. Instead, a conflict place intuitively captures a negative information flow forbidden by the policy. It is a place where transitions at different security levels L and H compete for a token in some computation. As observed above, each transition influence the other, and necessarily, since we work in a partially ordered domain, in one direction the influence is forbidden by the policy.

Definition 13 (conflict place). *Let H, $L \in \mathscr{L}$ be two security levels such that $H \not\rightsquigarrow L$. A place $p \in P$ of a net system $\mathbf{N}$ is HL-*conflict *if:*

(i) $p \in {}^\bullet l \cap {}^\bullet h$ for transitions $h, l \in T$ such that $\lambda h = H$ and $\lambda l = L$;

(ii) there exists $m \in [m_0\rangle$ satisfying $m[h\tau\rangle$, $m[\tau l\rangle$ and $\langle m[h\tau\rangle(p) < F(p,l)$, with $\tau \in T^*_{\overline{\uparrow H}}$.

Place p is called conflict *if it is HL-conflict for some $H, L \in \mathscr{L}$.*

In this case, the observation of l allows one to deduce that there was a time in the past at which h could have fired, but it did not. This determines an illegal information flow from level H to level L.

Lemma 1 (U-BNDC through HL-causal and HL-conflict places). *Let $U \subseteq \mathscr{L}$ be an upper set. A net system $\mathbf{N}$ is not U-BNDC iff $\mathbf{N}$ contains a HL-causal or HL-conflict place for some $H \in U$ and $L \in \overline{U}$.*

Proof. We build on the results for the two-level case in [7]. In fact, let $U \subseteq \mathscr{L}$ be an upper set. Relabel the net over a two-level domain $\mathscr{B} = \{Low \rightsquigarrow High\}$ by assigning all transitions in T_U security level *High* and all transitions in $T_{\overline{U}}$ security level *Low*. Then U-view bisimilarity in the original net is exactly $\{Low\}$-bisimilarity in the two-level net.

Now, assume that $\mathbf{N}$ is not U-BNDC. It follows that the relabeled net is not BNDC in the two-level domain $\mathscr{B}$. By [7, Theorem 3.3] there must be a causal or conflict place p in $\mathbf{N}$. We can conclude by observing that p is an HL-causal or HL-conflict place, respectively, for some $H \in U$ and $L \in \overline{U}$. For instance, assume that p is a causal place with respect to the two-level domain $\mathscr{B}$. This means that $p \in {}^\bullet l \cap h^\bullet$ for transitions $h, l \in T$ of levels *High* and *Low* respectively, and there exists $m \in [m_0\rangle$ satisfying $m[h\tau l\rangle$, $m[\tau\rangle$ and $\langle m[\tau\rangle(p) < F(p,l)$, with $\tau \in T^*_{Low}$. Therefore, with respect to the original labelling, $\lambda h = H$, $\lambda l = L$ with $H \in U$ and $L \in \overline{U}$ and $\tau \in T^*_{\overline{U}}$. Since U is an upper set and $H \in U$, we have that $\uparrow H \subseteq U$ and thus $\overline{U} \subseteq \overline{\uparrow H}$ which in turn implies $T_{\overline{U}} \subseteq T_{\overline{\uparrow H}}$. Hence $\tau \in T^*_{\overline{H}}$ and, therefore, p is a HL-causal place.

The converse implication works in an analogous way. □

Theorem 1 (BNDC through causal and conflict places). *A net system $\mathbf{N}$ is BNDC iff $\mathbf{N}$ does not contain any causal or conflict place.*

Proof. Immediate consequence of the definition of BNDC (Definition 11) and Lemma 1. □

As an example, consider the net system and the security domain in Fig. 1. The system is not BNDC as witnessed by places a_0, b_0 and *Free*, which are causal. E.g., for a_0 observe that $a_0 \in {}^\bullet get_A \cap upd1_C{}^\bullet$, with $C \not\rightsquigarrow A$. Moreover, if we consider the marking m reached after firing $get_A\, send_A$ (i.e., $m_0[get_A\, send_A\rangle m$), we have $m[upd1_C \,\epsilon\, get_A\rangle$ (in the notation of Definition 12, the firing sequence $\tau = \epsilon$, , i.e., it is empty) and $\langle m[\epsilon\rangle(a_0) = m(a_0) = 0 < 1 = F(a_0, get_A)$, i.e., the firing of $upd1_C$ is essential to enable get_A. Observe that *Free* is also a conflict place. In fact, $Free \in {}^\bullet send_A \cap {}^\bullet send_B$, $A \not\rightsquigarrow B$ and there is a reachable marking m, obtained by firing get_A and get_B (i.e., $m_0[get_A get_B\rangle m$) such that $m[send_A\rangle$, $m[send_B\rangle$ and $\langle m[send_A\rangle(Free) = 0 < F(Free, send_B)$, i.e., the firing of

get_A disables get_B (again, with respect to the notation of the general Definition 13, the sequence $\tau = \epsilon$). Intuitively, the origin of the conflict place is the mutually exclusive access to the cache by the sensors that determines a covert channel between them. The causal places, instead, are the result of the control switch between sensor and cache during an update. If the requisite is to ensure that when a value is sent to the cache, an update actually happens, these issues seem to be hardly solvable. The interference between the sensor seems to be unavoidable as well, unless a dedicated cache is added. In Section 4 we will see how by using intransitive policies and downgrading transitions these interferences can be amended.

As announced before, by exploiting the above characterization of BNDC we can prove that any security violation can be detected by analyzing only upper sets of the kind $U = \uparrow L$ for some level L. We first need a technical lemma.

Lemma 2 (BNDC is preserved by union). *Let* $\mathbf{N}$ *be a net system and let* $U_1, U_2 \subseteq \mathscr{L}$ *be upper sets. If* $\mathbf{N}$ *is* U_1*-BNDC and* U_2*-BNDC then it is* $U_1 \cup U_2$*-BNDC.*

Proof. We exploit the characterization of BNDC through causal and conflict places in Lemma 1. Assume that $\mathbf{N}$ is not $U_1 \cup U_2$-BNDC. Then by Lemma 1 the net system $\mathbf{N}$ contains a HL-causal or HL-conflict place p for some $H \in U_1 \cup U_2$ and $L \in \overline{U_1 \cup U_2} = \overline{U_1} \cap \overline{U_2}$. Therefore either $H \in U_1$ or $H \in U_2$, and L is in both complements. Again by Lemma 1, we deduce that either $\mathbf{N}$ is not U_1-BNDC or it is not U_2-BNDC, as desired. □

Notice that the converse implication does not hold: a net system can be $U_1 \cup U_2$-BNDC without being neither U_1-BNDC nor U_2-BNDC. The reason is that if a system contains a forbidden interference between levels in U_1 and U_2, this interference could not be observed with a $U_1 \cup U_2$ bisimulation. A simple example of this can be obtained by considering a domain $\mathscr{L} = \{\bot, A, B, AB\}$, with policy $\bot \rightsquigarrow A \rightsquigarrow AB$ and $\bot \rightsquigarrow B \rightsquigarrow AB$. Assume that there are forbidden interferences between A and B. Then if we take $U_1 = \{A, AB\}$ and $U_2 = \{B, AB\}$, we have that the system is $U_1 \cup U_2$-BNDC since all forbidden flows are internal to $U_1 \cup U_2$, but it is neither U_1-BNDC nor U_2-BNDC, due to the interferences between A and B.

The desired result then follows as a corollary.

Corollary 1 (multilevel BNDC to 2-level BNDC). *A net system* $\mathbf{N}$ *is BNDC iff* $\mathbf{N}$ *is* $\uparrow L$*-BNDC for each* $L \in \mathscr{L}$*.*

Another easy consequence of the characterization in Theorem 1 is that a BNDC net systems, remains BNDC if we take any reachable marking as initial marking. This and the subsequent characterization of BNDC will be useful later in the treatment of the intransitive case (see the proof of Proposition 2).

Lemma 3 (persistency of BNDC). *If a net system* $\mathbf{N} = (N, m_0)$ *is BNDC, then also* $\mathbf{N}' = (N, m)$ *is BNDC for all* $m \in [m_0\rangle$*.*

Proof. We prove the contrapositive. Let $m \in [m_0\rangle$ be a reachable marking in $\mathbf{N}$ and assume that $\mathbf{N}' = (N, m)$ is not BNDC. By Theorem 1, the net system $\mathbf{N}'$ contains a causal or conflict place p. Assume, without loss of generality, that it is causal and let $m' \in [m\rangle$ be the marking satisfying Definition 12. Then p is causal also in $\mathbf{N}$ because it satisfies all the structural conditions and $m' \in [m_0\rangle$. Hence, again by Theorem 1, the net system $\mathbf{N}$ is not BNDC, as desired. □

3.3. Characterizing Multilevel BNDC in Safe Net Systems

In this section we set the ground for the design of an algorithm for checking multilevel BNDC over safe Petri net systems. We start by showing that BNDC can be characterized in safe nets by relying on weaker, algorithmically more convenient, notions of causal and conflict place.

Notation 2. *Given a safe net system* $\mathbf{N}$ *we denote by* $t^- = \{p \in P : p \in {}^\bullet t \ \wedge \ p \notin t^\bullet\}$ *and* $t^+ = \{p \in P : t^\bullet \ \wedge \ p \notin {}^\bullet t\}$ *the sets of places where the firing of* t *decreases and increases, respectively, the number of tokens.*

Definition 14 (weak causal place). *Let* $H, L \in \mathscr{L}$ *be two security levels such that* $H \not\rightsquigarrow L$ *and let* $\mathbf{N}$ *be a net system. A* weak HL-causal place *in* $\mathbf{N}$ *is any place* $p \in {}^\bullet l \cap h^+$*, for some* $l, h \in T$ *such that* $\lambda h = H$*,* $\lambda l = L$*, and there exists a reachable marking* $m \in [m_0\rangle$ *such that* $m[h\tau l\rangle$*, with* $\tau \in T^*$*. Place* p *is called* weak causal *if it is a weak* HL*-causal for some* $H, L \in \mathscr{L}$*.*

Intuitively, the existence of a firing sequence $h\tau l$ and of the place $p \in {}^\bullet l \cap h^+$ gives the possibility that the firing of l depends on the firing of h, thus determining an illegal flow from level $H = \lambda h$ to level $L = \lambda l$. We will prove that for safe nets this potential flow actually exists. Weak conflict places are defined along the same lines.

Definition 15 (weak conflict place). *Let* $H, L \in \mathscr{L}$ *be two security levels such that* $H \not\rightsquigarrow L$ *and let* $\mathbf{N}$ *be a net system. A* weak HL-conflict place *in* $\mathbf{N}$ *is any place* $p \in {}^\bullet l \cap h^-$*, for some* $l, h \in T$ *such that* $\lambda h = H$*,* $\lambda l = L$*, and there exists a reachable marking* $m \in [m_0\rangle$ *such that* $m[h\rangle$ *and* $m[\tau l\rangle$*, with* $\tau \in T^*$*. Place* p *is called* weak conflict *if it is a weak* HL*-conflict for some* $H, L \in \mathscr{L}$*.*

As suggested by the terminology it is immediate to see that that any causal/conflict place is a weak causal/conflict place, while the converse does not hold. However, for safe nets from the presence of a weak causal or weak conflict place we can deduce the presence of a (possibly different) causal or conflict place. Hence they witness the failure of BNDC.

Theorem 2 (BNDC through weak causal and conflict places). *Let* $\mathbf{N}$ *be a safe net system. Then* $\mathbf{N}$ *is not BNDC iff* $\mathbf{N}$ *has either a weak causal place or a weak conflict place.*

Proof. As in Lemma 1, we can build on the results for the two-level case in [7]. If $\mathbf{N}$ is not BNDC in the multilevel domain $\mathscr{L}$, then, by definition (Definition 11) it is not U-BNDC for some upper set $U \subseteq \mathscr{L}$. Consider the net system $\mathbf{N}'$ obtained by relabeling $\mathbf{N}$ over the two-level domain $\mathscr{B} = \{Low \rightsquigarrow High\}$: transitions in T_U and $T_{\overline{U}}$ are assigned security level *High* and *Low*, respectively. Then, as observed in the proof of Lemma 1, the two-level net $\mathbf{N}'$ is not BNDC. Therefore, by Theorem [7, Theorem 3.10], $\mathbf{N}'$ contains a weak causal or conflict place p. We can conclude by observing that in $\mathbf{N}$ place p is weak HL-causal or weak HL-conflict, respectively, for some $H \in U$ and $L \in \overline{U}$. In order to see this, assume, for instance, that p is a weak causal place in the two-level net $\mathbf{N}'$. This means that $p \in {}^\bullet l \cap h^+$ for transitions $h, l \in T$ of levels *High* and *Low* respectively, and there exists $m \in [m_0\rangle$ satisfying $m[h\tau l\rangle$, with $\tau \in T^*$. If, in the original labelling, $\lambda h = H$ and $\lambda l = L$ then $H \in U$ and $L \in \overline{U}$, and since U is an upper set $H \not\rightsquigarrow L$. Therefore p is a weak HL-causal place. Similarly, if p is a weak conflict place in $\mathbf{N}'$, we can show that p is a weak HL-conflict place in $\mathbf{N}$.

The converse implication works in an analogous way. □

3.4. Characterizing Multilevel BNDC in the Unfolding of Safe Nets

We next observe that occurrences of weak causal and conflict places in the unfolding of a safe net system can be given a characterization in terms of structural conditions. Jointly with Theorem 2, this leads to a characterization of the BNDC property on the unfolding of safe nets.

Notation 3. *For a condition b and an event e in the unfolding $\mathcal{U}(\mathbf{N})$ we set $e^+ = \{b \in P^{(\omega)} : \pi_1(b) \in \pi_1(e)^+\}$ and $e^- = \{b \in P^{(\omega)} : \pi_1(b) \in \pi_1(e)^-\}$.*

Lemma 4 (weak causal/conflict places in the unfolding). *Let $\mathbf{N}$ be a net system and let p be a place in $\mathbf{N}$. Then*

(i) p is a weak causal place in $\mathbf{N}$ iff there are events h', l' with $\lambda h' \not\rightarrow \lambda l'$ and a condition $b \in {}^\bullet l' \cap h'^+$ such that $\pi_1(b) = p$;

(ii) p is a weak conflict place in $\mathbf{N}$ iff there are events h', l' with $\lambda h' \not\rightarrow \lambda l'$ and $[h') \cup [l'] \in \mathcal{C}(\mathcal{U}(\mathbf{N}))$, and a condition $b \in {}^\bullet l' \cap h'^-$ such that $\pi_1(b) = p$.

Proof. (i) Let p be a weak HL-causal place in $\mathbf{N}$, for some levels $H, L \in \mathscr{L}$ such that $H \not\rightarrow L$. Then there are transitions h, l in $\mathbf{N}$ such that $\lambda l = L$, $\lambda h = H$ and a place $p \in {}^\bullet l \cap h^+$ such that there exist $m \in [m_0\rangle$ and $\tau \in T^*$ such that $m[h\tau l\rangle$. Therefore in the unfolding there are occurrences of p, l, h, namely a condition b and events l', h' such that $\pi_1(b) = p$, $\pi_1(h') = h$, $\pi_1(l') = l$ and $b \in {}^\bullet l' \cap h'^+$, as desired.

Vice versa, assume that $h = \pi_1(h')$, $l = \pi_1(l')$ and $p = \pi_1(b)$, for some condition b and events l', h' such that $\lambda h' = H \not\rightarrow L = \lambda l'$. Since, by hypotheses, $b \in {}^\bullet l' \cap h'^+$, we deduce that $p \in {}^\bullet l \cap h^+$. Consider the markings $m_1 = \mathsf{M}([h'))$ and $m_2 = \mathsf{M}([l'))$ and let τ be any linearization of $[l') \setminus [h']$ compatible with causality. Then $m_1[h\tau\rangle m_2[l\rangle$ and hence p is a weak HL-causal place in $\mathbf{N}$.

(ii) Let p be a weak HL-conflict place in $\mathbf{N}$, for some levels $H, L \in \mathscr{L}$ such that $H \not\rightarrow L$. Then $p \in {}^\bullet l \cap h^-$ and there exists $m \in [m_0\rangle$ and $\tau \in T^*$ such that $m[h\rangle$ and $m[\tau l\rangle$. Therefore in the unfolding there are occurrences of p, l, h, namely a condition b and events l', h' such that $\pi_1(b) = p$, $\pi_1(h') = h$, $\pi_1(l') = l$, with $b \in {}^\bullet l' \cap h'^-$. Moreover there is a configuration C such that $\mathsf{M}(C) = \langle m[\tau l\rangle$ and $C \supseteq [h') \cup [l']$. Hence $[h') \cup [l'] \in \mathcal{C}(\mathcal{U}(\mathbf{N}))$.

Vice versa, assume that $h = \pi_1(h')$, $l = \pi_1(l')$ and $p = \pi_1(b)$, for some condition b and events l', h' such that $\lambda h' = H \not\rightarrow L = \lambda l'$. Since, by hypotheses, $b \in {}^\bullet l' \cap h'^-$, we deduce that $p \in {}^\bullet l \cap h^-$. Consider the markings $m_1 = \mathsf{M}([h'))$ and $m_2 = \mathsf{M}([l'))$ and let τ be any linearization of $[l') \setminus [h']$ compatible with causality. Then $m_1[h\rangle$ and $m_1[\tau\rangle m_2[l\rangle$, the second firing sequence being possible since $[h') \cup [l'] \in \mathcal{C}(\mathcal{U}(\mathbf{N}))$. Hence p is a weak HL-conflict place in $\mathbf{N}$. □

As an immediate consequence we obtain the following characterization that relates BNDC to the absence, in the unfolding, of conditions enjoying certain simple structural properties.

Corollary 2 (BNDC in the unfolding of safe nets). *Let $\mathbf{N}$ be a safe net system. Then $\mathbf{N}$ is not BNDC iff there exist events h', l' such that $\lambda h' \not\rightarrow \lambda l'$ and a condition b in $\mathcal{U}(\mathbf{N})$ such that one of the following holds:*

(i) $b \in {}^\bullet l' \cap h'^+$

(ii) $b \in {}^\bullet l' \cap h'^-$ *and* $[h') \cup [l'] \in \mathcal{C}(\mathcal{U}(\mathbf{N}))$.

Proof. Direct consequence of Theorem 2 and Lemma 4. □

An interesting fact is the possibility of reducing, by a suitable transformation of the net, all interferences to causal ones. This was possible in the two-level case and, nicely, the construction can be adapted to work with general multilevel domains. It will be useful for the development of an efficient unfolding-based algorithm for checking BNDC (see § 3.5) since occurrences of causal places are characterized by a very local condition in the unfolding, while checking the condition for weak conflict places requires an exploration of the history of the interacting transitions. It will play a role also in the algorithm for checking intransitive non-interference (see § 4.5).

Definition 16 (causal reduct). *Given a net system* $\mathbf{N}$*, let* $T_\# = \{h \in T \mid \exists l \in T \,.\, {}^\bullet l \cap h^- \neq \emptyset \ \wedge \ \lambda h \not\leadsto \lambda l\}$ *and define the* causal reduct $\gamma(\mathbf{N})$ *as the net system* (N', m'_0) *obtained from* $\mathbf{N}$ *as follows:*

- $P' = P \cup \{p_H\} \cup \{p_h \mid h \in T_\#\}$
- $T' = T \cup \{c_h \mid h \in T_\#\} \cup T_\gamma$ *where* T_γ *is defined as:*
  $T_\gamma = \{c_{lh} \mid h \in T_\#$ *and* $l \in T$ *and* $\lambda h \not\leadsto \lambda l$ *and* ${}^\bullet l \cap h^- \neq \emptyset\}$
- $\lambda c_h = \lambda h \quad \forall h \in T_\#$
  $\lambda c_{lh} = \lambda l \quad \forall c_{lh} \in T_\gamma$
- $F'(x,y) = F(x,y) \quad \forall x, y \in P \cup T$
  $F'(p, c_h) = F'(c_h, p) = F(p,h) \quad \forall h \in T_\#$ *and* $\forall p \in P$
  $F'(c_h, p_h) = 1$ *and* $F'(p_H, c_h) = 1 \quad \forall h \in T_\#$
  $F'(p, c_{lh}) = F(p,l)$ *and* $F'(c_{lh}, p) = F(l,p) \quad \forall c_{lh} \in T_\gamma$ *and* $\forall p \in P$
  $F'(p_h, c_{lh}) = 1 \quad \forall c_{lh} \in T_\gamma$
- $m'_0(p) = m_0(p) \quad \forall p \in P$
  $m'_0(p_h) = 0 \quad \forall h \in T_\#$
  $m'_0(p_H) = 1$

The set $T_\#$ includes the transitions that have in their pre-set a potential conflict place, i.e., transitions h for which there is some other transition l with $\lambda h \not\leadsto \lambda l$ such that ${}^\bullet l \cap h^- \neq \emptyset$. For each transition $h \in T_\#$ we add to the net another transition c_h that tests whether h is enabled, by consuming and reproducing the pre-set of h. Transition c_h additionally produces a token in a new place p_h. Hence, the presence of a token in p_h witnesses that the fact that previously h has been enabled. All new transitions c_h inputs a token from a new common place p_H, ensuring that each firing sequence will include at most one of them. Additionally, for each $h \in T_\#$ and for all potentially illegal conflicts, i.e., for each l such that $h^- \cap {}^\bullet l \neq \emptyset$ and $\lambda h \not\leadsto \lambda l$, we insert a new transition c_{lh} that is a copy of l, but in addition it inputs the token generated by c_h in p_h. In this way, the execution of c_{lh} will be possible only if in the past h was enabled but not executed, thus, intuitively, transforming a conflict interference into a causal one.

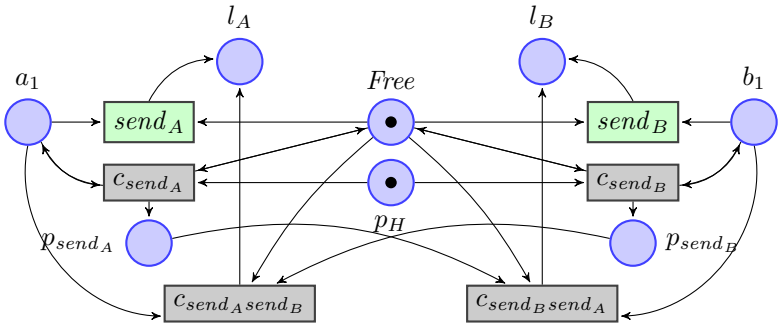


Figure 5: A part of the causal reduct of the running example $\mathbf{S}$. The rest of the net is unchanged. Green transitions belong to the original net system.

As an example, (a part of) the causal reduct of the running example net system $\mathbf{S}$ is reported in Fig. 5. Since $A \not\rightsquigarrow B$ and $Free \in {}^\bullet send_B \cap send_A{}^-$ we have that $send_A \in T_\#$. This leads to the introduction of the transitions c_{send_A} and $c_{send_A send_B}$ connected by place p_{send_A}. Since *Free* is weak conflict place p_{send_A} becomes a weak causal place.

We define two functions $f_1 : (T')^* \to T^*$ and $f_2 : [m'_0\rangle \to [m_0\rangle$ which project firings sequences and markings of $\gamma(\mathbf{N})$ over $\mathbf{N}$ by stripping the newly added items:

- $f_1(\sigma) = \begin{cases} \varepsilon & \text{if } \sigma = \varepsilon \\ t f_1(\sigma') & \text{if } \sigma = t\sigma',\ t \in T \\ f_1(\sigma') & \text{if } \sigma = c_h\sigma',\ h \in T_\# \\ l f_1(\sigma') & \text{if } \sigma = c_{lh}\sigma',\ c_{lh} \in T_\gamma \end{cases}$
- $f_2(m')$ is the restriction of m' to P.

It is easy to see that functions f_1 and f_2 are a indeed a simulation of $\gamma(\mathbf{N})$ into $\mathbf{N}$, in the following sense.

Lemma 5 (simulation). *Let $\mathbf{N}$ be a net system and let $f_1 : (T')^* \to T^*$ and $f_2 : [m'_0\rangle \to [m_0\rangle$ be functions defined as above. If $m'_1[\tau\rangle m'_2$ in the causal reduct $\gamma(\mathbf{N})$, then $f_2(m'_1)[f_1(\tau)\rangle f_2(m'_2)$ in the original net system $\mathbf{N}$.*

Proof. Straightforward induction on $|\tau|$.

Proposition 1 (BNDC in the causal reduct). *Let $\mathbf{N}$ be a net system. Then $\mathbf{N}$ is not BNDC iff $\gamma(\mathbf{N})$ contains a weak causal place.*

Proof. Let $\mathbf{N}$ be a net system and let $\gamma(\mathbf{N}) = (N', m'_0)$ be its causal reduct. In the proof we use the notation of Definition 16.

($\Rightarrow$) If $\mathbf{N}$ is not BNDC, then it contains either a (weak) causal place or a (weak) conflict place. Let $p \in {}^\bullet l \cap h^+$ be a weak causal place in $\mathbf{N}$. Then there exists a reachable marking $m \in [m_0\rangle$ satisfying the conditions of Definition 14. The unique marking m' on

$\mathbf{N}'$ that extends m and such that $m'(p') = m'_0(p')$ for $p' \in P' \setminus P$ witnesses the fact that p is a weak causal place in $\gamma(\mathbf{N})$.

Let $p \in {}^\bullet l \cap h^-$ be a weak conflict place in $\mathbf{N}$. There exist a reachable marking $m \in [m_0\rangle$ and a sequence $\tau \in T^*$ such that $m[h\rangle$, $m[\tau l\rangle$. Let m' be the unique marking on N' that agrees with m on P and agrees with m'_0 on $P' \setminus P$. Then by construction $m'[c_h \tau c_{lh}\rangle$ and $p_h \in {}^\bullet c_{lh} \cap {c_h}^+$, meaning that p_h is a weak causal place in $\gamma(\mathbf{N})$.

($\Leftarrow$) Assume that p' is a weak causal place in $\gamma(\mathbf{N})$. Therefore there are $h', l' \in T'$, $p' \in {}^\bullet l' \cap {h'}^-$, and $m' \in [m'_0\rangle$ such that $\lambda h' \not\leadsto \lambda l'$, and $m'[h'\tau l'\rangle$ with $\tau \in T'^*$. We distinguish various cases according to the membership of h' and l'.

- If $h' \in T$ then necessarily $p' \in P$ and, by Lemma 5, there is a firing sequence $f_2(m')[h' f_1(\tau') f_1(l')\rangle$ in $\mathbf{N}$. Now, note that if $l' \in T$ then $f_1(l') = l'$, otherwise, it must be $l' \in T_\gamma$ and thus, also in this case, $f_1(l') \neq \epsilon$. Hence in both cases p' is a causal place in $\mathbf{N}$ involving h' and $f_1(l')$.
- If $h' \in T' \setminus T$ then, by construction, it must be $l' \in T' \setminus T$, with $h' = c_h$, $l' = c_{lh}$ and $p' = p_h$ for suitable $l, h \in T$. By Lemma 5 and observing that $f_1(c_h \tau' c_{lh}) = f_1(\tau') l$, we deduce $f_2(m')[f_1(\tau') l\rangle$. Moreover, since $m'[c_h\rangle$, we have that $f_2(m')[h\rangle$ (recall that, apart from place p_H, transitions c_h and h have the same pre-set). Finally, ${}^\bullet l \cap h^- \neq \emptyset$, otherwise c_{lh} would have not been inserted. Hence any $p \in {}^\bullet l \cap h^-$ is a weak conflict place in $\mathbf{N}$. □

3.5. Unfolding-based Algorithm for Multilevel BNDC on Safe Nets

Relying on the theory developed so far, we provide an algorithm for checking multilevel BNDC on a finite prefix of the unfolding of a safe net. Interestingly, while the definition of multilevel BNDC is formulated in terms of a number of checks in the two-level setting, we propose a verification technique based on the construction of a single unfolding prefix.

3.5.1. Complete prefixes for multilevel BNDC interferences.

Starting with [18] techniques have been developed for efficiently constructing finite prefixes of the unfolding which are complete with respect to some property of interest. As already discussed in [7] in the two-level case, a prefix which is complete for marking reachability could omit relevant information concerning interferences. In order to exploit the theory of finite prefixes in [19], as a first step we identify a completeness criterion ensuring that a prefix includes at least a representative for causal interferences, when a net system is not multilevel BNDC.

Intuitively, in order to be complete for (causal) interferences a prefix should include representatives of all possible situations of direct causal dependency. For this aim, in the two-level case, markings were enriched by recording which tokens were generated by high level transitions. Here we need to record, for each token, the level of the transition generating it. The notion of completeness is adapted accordingly.

Definition 17 (c-marking, c-complete prefix). *Let* $\mathbf{N}$ *be a net system and let* $C \in \mathcal{C}(\mathcal{U}(\mathbf{N}))$. *The* confidentiality marking (c-marking) *of* C *is* $\mathsf{M}^*(C) = \langle \mathsf{M}(C), \Lambda_C \rangle$, *where* $\Lambda_C : P \to 2^{\mathscr{L}}$ *is the function defined as:*

$$\Lambda_C(p) = \{\lambda e \mid e \in T^{(\omega)} \wedge p \in \pi_1(e^+ \cap C^\circ)\}$$

A prefix $\mathcal{U}_F$ of $\mathcal{U}(\mathbf{N})$ is complete for c-marking reachability*, or simply* c-complete*, when for any configuration $C \in \mathcal{C}(\mathcal{U}(\mathbf{N}))$ there exists $C' \in \mathcal{C}(\mathcal{U}_F)$ such that $\mathsf{M}^*(C) = \mathsf{M}^*(C')$.*

In words, the marking of a configuration C is enriched with a function Λ_C, which records, for each place, the security level of the transition, if any, that generated the token in that place. Observe that, since nets are safe, for any $p \in P$ there is at most one token and thus at most one event e such that $p \in \pi_1(e^+ \cap C^\circ)$. This means that $\Lambda_C(p)$ is empty or a singleton, and thus Λ_C can be seen as a partial function from P to $\mathscr{L}$.

The unfolding prefix is generated, inductively, starting from the initial marking and adding, at each step, an occurrence of a transition enabled by a marking in the current prefix. We next formalize the notion of possible extension of a prefix.

Definition 18 (possible extensions). *Let $\mathbf{N}$ be a net systems and let $\mathcal{U}_F$ be a prefix of $\mathcal{U}(\mathbf{N})$. A* possible extension *of $\mathcal{U}_F$ is any event $e \in T^{(\omega)}$ such that $e \notin T_{\mathcal{U}_F}$ and ${}^\bullet e \subseteq C^\circ$ for some configuration C of $\mathcal{U}_F$. The set of possible extensions of $\mathcal{U}_F$ is denoted $PE(\mathcal{U}_F)$. The* one-step extension *of U, denoted $U^\triangleright$, is the prefix obtained by adding to $\mathcal{U}_F$ all possible extensions in $PE(\mathcal{U}_F)$.*

We can now show that a c-complete prefix $\mathcal{U}_F$ of $\mathcal{U}(\mathbf{N})$ includes sufficient information for deciding whether or not $\mathbf{N}$ contains a weak causal place. Actually, besides the events in the prefix, we need to consider also its possible extensions. In the algorithmic procedure for producing the prefix, such events, that would be added by a further unfolding step, are indeed added and marked as cut-offs.

Theorem 3 (weak causal places in c-complete prefixes). *Let $\mathbf{N}$ be a safe net system and let $\mathcal{U}_F$ be a c-complete prefix of $\mathcal{U}(\mathbf{N})$. Then p is a weak causal place in $\mathbf{N}$ iff there exist in ${\mathcal{U}_F}^\triangleright$ a condition b and events h' and l' such that $\pi_1(b) = p$, $b \in {}^\bullet l' \cap {h'}^+$ and $\lambda h' \not\leftrightarrow \lambda l'$.*

Proof. Let p be a causal place in $\mathbf{N}$. By Lemma 4(i) in $\mathcal{U}(\mathbf{N})$ there are a condition b' and events h'', l'' such that $b' \in {}^\bullet l'' \cap {h''}^+$, $\lambda h'' \not\leftrightarrow \lambda l''$, and $\pi_1(b') = p$.

Let $C' = [l'')$. By c-completeness of $\mathcal{U}_F$ there exists a configuration $C \in \mathcal{C}(\mathcal{U}_F)$ such that $\mathsf{M}^*(C) = \mathsf{M}^*(C')$. In particular $\Lambda_C(p) = \Lambda_{C'}(p) \ni \lambda h'$. Thus we deduce the existence of h' and b such that $\lambda h' = \lambda h''$, $b \in {h'}^+ \cap C^\circ$ and $\pi_1(b) = p$. Since $\mathsf{M}(C) = \mathsf{M}(C')$ and $\mathsf{M}(C')[\pi_1(l'')\rangle$ there exists l' in ${\mathcal{U}_F}^\triangleright$ such that $\pi(l') = \pi_1(l'')$ hence $\lambda(l') = \lambda(l'')$. Moreover, since $p \in {}^\bullet\pi_1(l')$ and $\pi_1(b) = p$ we also have $b \in {}^\bullet l'$. Summing up, $\lambda(h') = \lambda(h'') \not\leftrightarrow \lambda(l'') = \lambda(l')$ and $b \in {}^\bullet l' \cap {h'}^+$, as desired.

For the converse implication, just observe that ${\mathcal{U}_F}^\triangleright$ is a subnet of $\mathcal{U}(\mathbf{N})$ and use Lemma 4(i). □

Combining the previous theorem with Proposition 1 and Lemma 4 we obtain that one can check BNDC on a c-complete prefix of the unfolding of its causal reduct.

Corollary 3 (BNDC on c-complete prefixes). *Let $\mathbf{N}$ be a safe net system and let $\mathcal{U}_F$ be a c-complete prefix of $\mathcal{U}(\gamma(\mathbf{N}))$. Then $\mathbf{N}$ is not BNDC iff there exist events $h', l' \in {\mathcal{U}_F}^\triangleright$ such that $\lambda h' \not\leftrightarrow \lambda l'$ and ${}^\bullet l' \cap {h'}^+ \neq \emptyset$.*

3.5.2. Unfolding-based algorithm for multilevel BNDC on safe net systems.

From Corollary 3 we can derive an algorithm for checking the BNDC property on safe nets. It incrementally constructs a c-complete prefix of the unfolding of the causal reduct, checking, at any step, for the presence of a condition witnessing an illegal flow (namely, a condition produced by an event at level H and consumed by an event at level L such that $H \not\rightsquigarrow L$). Since safe nets are clearly finite-state, a c-complete prefix is finite and thus the process necessarily terminates.

The construction of the prefix stops at events, called cut-offs [18], that are "useless" since they produce a c-marking already produced by another event with smaller history. The term smaller refers to some chosen adequate order $\prec$ on configurations [19]. The simplest option is to define $C \prec C'$ is $|C| < |C'|$ as in [18]. The size of a complete prefixes can be considerably reduced by choosing finer adequate orders [22]. This is actually done in the tool MultiUBIC.

Definition 19 (cut-off). *Let* $\mathbf{N}$ *be a net system and let* $\mathcal{U}_F$ *be a prefix of* $\mathcal{U}(\mathbf{N})$*. An event* e *in* $\mathcal{U}_F$ *is called a* cut-off *when there exists another event* e' *in* $\mathcal{U}_F$ *such that* $\mathsf{M}^*([e]) = \mathsf{M}^*([e'])$ *and* $[e'] \prec [e]$.

The complete prefix is created by selecting iteratively one possible extension at a time, the starting point being a prefix consisting only of the initial marking of the original net. The construction of the prefix proceeds by adding events with $\prec$-minimal history and stopping at cut-offs. From the general theory in [19], one can deduce that for a safe net the algorithm stops producing a finite c-complete prefix. We omit the details as they largely overlap with those for the two-level case in [7].

On these bases we developed the algorithm for checking BNDC on safe net systems outlined in Fig. 6. It first computes the causal reduct $\gamma(\mathbf{N})$ of $\mathbf{N}$. Then it builds a c-complete prefix of the unfolding of $\gamma(\mathbf{N})$, looking, at each step, for the presence of direct causalities between events satisfying the conditions in Theorem 3.

Corollary 4 (correctness of the algorithm for safe nets). *Let* $\mathbf{N}$ *be a safe net system. Then the algorithm of Fig. 6 always terminates and provides the answer 'yes' iff* $\mathbf{N}$ *is BNDC.*

Proof. Immediate consequence of Corollary 3. □

A bound on the size of a complete prefix can be obtained along the lines of [22]. When the adequate order $\prec$ is total, in the worst case the number of events in a c-complete prefix coincides with the number of c-markings (observe that given distinct events e, e' such that $\mathsf{M}^*([e]) = \mathsf{M}([e'])$ then, since the order is total, either $[e] \prec [e']$ or vice versa and thus one of the events is a cut-off). In turn, the number of c-markings is $(|\mathscr{L}|+1)^{|S|}$ since each place can either be marked or not, and, for marked places, we record the security level of the transition that generated the token, that gives $|\mathscr{L}| + 1$ possibilities for each place.

Data: A safe net system $\mathbf{N}$.
compute $\gamma(\mathbf{N})$
$\mathcal{U}_F = \gamma(m_0)$
$pe = PE(\mathcal{U}_F)$
while $pe \neq \emptyset$ **do**
  take $e \in pe$ such that $[e]$ is $\prec$-minimal;
  if $\exists h, b \in \mathcal{U}_F.\ \lambda h \not\rightsquigarrow \lambda e \wedge b \in {}^\bullet e \cap h^+$ **then**
    return *'no'*
  end
  add e to $\mathcal{U}_F$
  if *e is not a cut-off* **then**
    add $e^\bullet$ to $\mathcal{U}_F$
  end
  $pe = PE(\mathcal{U}_F)$
end
return *'yes'*;

Figure 6: Algorithm to decide BNDC on safe net systems.

4. Intransitive Multilevel Non-Interference

In this section we focus on intransitive policies. The general idea is that some flows of information between levels that cannot communicate directly become allowed if they are mediated by a chain of trusted intermediaries.

4.1. Bisimilarity-based Intransitive Non-Interference

In order to formalize the notion of violation with respect to an intransitive security policy, we resort to an the idea that resembles that of separability in [21]. Roughly speaking, in order to check whether there are illegal flows from a set of levels U, we artificially isolate that set by removing from the system all legitimate targets, namely elements whose level L' is such that $L \rightsquigarrow L'$ for some $L \in U$. Then we check if some level in U can still influence other levels in the rest of the system. If this happens, the influence is certainly illegal, as it cannot be mediated by a chain of legal intermediaries since any such chain has been broken by our construction. We extend the strict targets (Definition 2) to sets of levels $U \subseteq \mathscr{L}$ by defining $\Uparrow U = \uparrow U \setminus U$.

Definition 20 (U-BINI). *Let $U \subseteq \mathscr{L}$ be a set of levels. A net system $\mathbf{N}$ is U-BINI if for every reachable marking $m \in [m_0\rangle$ the net system $(N \setminus T_{\Uparrow U}, m)$ is U-BNDC in the domain $\mathscr{L}' = (\mathscr{L} \setminus \Uparrow U, \rightsquigarrow^*)$.*

Observe that the definition is well-given since, by construction, U is an upper set in the transitive domain $\mathscr{L}' = (\mathscr{L} \setminus \Uparrow U, \rightsquigarrow^*)$, hence it respects the requirements in Definition 11.

The definition can be understood as follows. As mentioned above, for a set of levels U we consider the net $N \setminus T_{\Uparrow U}$, obtained by pruning the transitions whose level is in $\Uparrow U$. These are the transition to which a flow from some level in U is admitted. The absence of an illegal flow from U is thus reduced to the absence of any flow in the pruned subsystem.

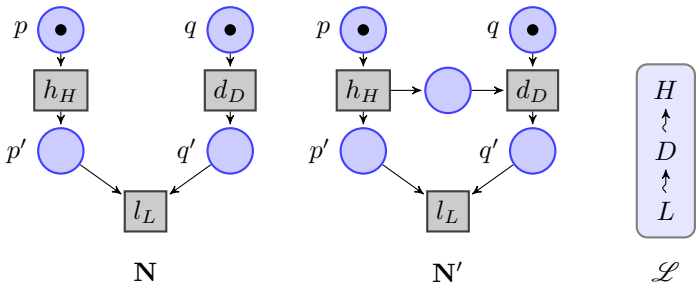


Figure 7: Over the security domain $\mathscr{L}$, net system $\mathbf{N}$ is not BINI while system $\mathbf{N}'$ is.

Note that an illegal flow from U could occur at any reachable marking m of the original system, but clearly the pruning operation can make m unreachable from the initial marking. This is the reason why the pruned net $N \setminus T_{\nuparrows U}$ needs to be checked with respect to any marking reachable in the original net system $\mathbf{N}$.

The fact that considering only the initial marking of the original net is not sufficient to identify all violations is exemplified by the net system $\mathbf{N}$ in Fig. 7. It is easy to see that the system has a violation of the policy $\lambda h \rightsquigarrow \lambda d \rightsquigarrow \lambda l$ witnessed by place p'. However in the system were level λd has been pruned, if we start from the initial marking, there is no way of firing l and thus of observing the interference. Instead, the interference is revealed when starting from marking $\{p, q'\}$, which was reachable in the original system.

The definition of BINI in a multilevel setting, at this point, is the natural one.

Definition 21 (BINI). *A net system* $\mathbf{N}$ *is BINI if it is* U*-BINI for every* $U \subseteq \mathscr{L}$*.*

In the next subsection we will show that that in a transitive security domain BINI coincides with BNDC, a result that confirms the view of BINI as a natural generalization of BNDC to intransitive domains.

4.2. Characterizing Multilevel BINI through Causal and Conflict Places

We show that the BINI property can be characterized, for general P/T nets, in terms of the absence of causal and conflict places. Interestingly enough, the notion of causal and conflict places remains formally the same as in the transitive case (see Definition 12 and Definition 13), but it is now taken in an intransitive security domain.

Lemma 6 (U**-BINI through** HL**-causal and** HL**-conflict places).** *Let* $\mathbf{N}$ *be a net system and let* $U \subseteq \mathscr{L}$ *be a set of levels. If* $\mathbf{N}$ *is not* U*-BINI then it contains a* HL*-causal or* HL*-conflict place for some* $H \in U$ *and* $L \notin U$*. Vice versa, if* $\mathbf{N}$ *contains a* HL*-causal or* HL*-conflict place for* $H, L \in \mathscr{L}$ *then* $\mathbf{N}$ *is not* $\{H\}$*-BINI.*

Proof. Let $U \subseteq \mathscr{L}$ be a set of levels. Let us first prove that if $\mathbf{N}$ is not U-BINI then $\mathbf{N}$ contains HL-causal or HL-conflict places with $H \in U$ and $L \notin U$. If $\mathbf{N}$ is not U-BINI then there is a marking $m \in [m_0\rangle$ for which $(\mathbf{N} \setminus T_{\nuparrows U}, m)$ is not U-BNDC in the transitive domain $\mathscr{L}' = (\mathscr{L} \setminus \nuparrows U, \rightsquigarrow^*)$. By Lemma 1, this implies that in the pruned net $(\mathbf{N} \setminus T_{\nuparrows U}, m)$ in the transitive domain $\mathscr{L}'$ there exists a HL-causal or HL-conflict place,

for some $H \in U$ and $L \in \mathscr{L}' \setminus U = (\mathscr{L} \setminus \Uparrow U) \setminus U = \mathscr{L} \setminus \uparrow U$. Spelling out the definition of causal place (Definition 12), it is readily seen that p is a HL-causal or HL-conflict place also in $\mathbf{N}$ with respect to the intransitive domain $\mathscr{L}$. In fact, $H \not\rightsquigarrow L$ since, as observed above, $L \in \mathscr{L} \setminus \uparrow U$. Moreover, the firing sequence τ required by the definition is included in $T_{\overline{\uparrow H}}$ also in $\mathscr{L}$. In fact, $\overline{\uparrow H} \subseteq \Uparrow U \cup (\uparrow H \cap U)$ and note that τ it does not include transitions with label in $\Uparrow U$, which has been removed by the pruning, and it does not include transitions in $\uparrow H \cap U$ since these would be in $\uparrow H$ also in the transitive domain $\mathscr{L}'$.

For the converse implication, assume that the net system $\mathbf{N}$ contains a HL-causal or HL-conflict place p for some $H, L \in \mathscr{L}$. Let m and τ be respectively the marking and firing sequence required by Definition 12 or Definition 13. It is easy to see that $(\mathbf{N} \setminus T_{\Uparrow\{H\}}, m)$ has p as HL-causal or HL-conflict place in the transitive domain $\mathscr{L}' = (\mathscr{L} \setminus \Uparrow\{H\}, \rightsquigarrow^*)$, because τ is not affected by the pruning. Hence, by Lemma 1, such net is not $\{H\}$-BNDC and thus $\mathbf{N}$ is not $\{H\}$-BINI. □

Theorem 4 (BINI through causal and conflict places). *A net system* $\mathbf{N}$ *is BINI iff* $\mathbf{N}$ *does not contain any causal or conflict place.*

Proof. Immediate consequence of Lemma 6. □

The characterization above is used to prove that in a transitive security domain the properties BINI and BNDC coincide. Besides being useful from a at a technical level, this result confirms that BINI is a natural generalization of BNDC.

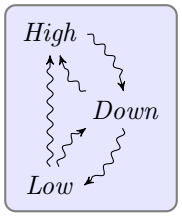
Proposition 2 (BINI is BNDC on transitive domains). *In a transitive multilevel security domain* $\mathscr{L}$*, a net system* $\mathbf{N}$ *is BINI if and only if it is BNDC.*

Proof. ($\Rightarrow$) We proceed by contradiction. Suppose that $\mathbf{N}$ is BNDC but not BINI. Then there are an upper set $U \subseteq \mathscr{L}$ and a reachable marking $m \in [m_0\rangle$ such that $(N \setminus T_{\Uparrow U}, m)$ is not U-BNDC over the domain $\mathscr{L}' = (\mathscr{L} \setminus \Uparrow U, \rightsquigarrow^*)$. By Theorem 1 such net contains a HL-causal place p with $H \in U$and $L \in \overline{U}$. Observe that since the complement is taken in $\mathscr{L}'$, we have that $H \not\rightsquigarrow L$. By definition of causal place, there are $p \in {}^\bullet l \cap h^\bullet$ for transitions $h, l \in T$ such that $\lambda h = H$ and $\lambda l = L$ and there is a marking $m' \in [m\rangle$, reachable in $(N \setminus T_{\Uparrow U}, m)$, satisfying $m'[h\tau l\rangle$, $m'[\tau\rangle$ and $\langle m[\tau\rangle(p) < F(p, l)$, with $\tau \in T^*_{\overline{\uparrow H}}$. Observe that since m is reachable in $\mathbf{N}$ also m' is reachable in $\mathbf{N}$. Moreover, $\overline{\uparrow H}$ in $\mathscr{L}'$ is a subset of $\overline{\uparrow H}$ in $\mathscr{L}$. Hence p is also a HL-place for $\mathbf{N}$ in $\mathscr{L}$. Therefore by Theorem 4 we conclude that $\mathbf{N}$ is not BINI.

($\Leftarrow$) Suppose for the sake of contradiction that $\mathbf{N}$ is BINI but not BNDC. Since it is not BNDC, by Theorem 1 there is a causal or conflict place p in $\mathbf{N}$.

Assume, e.g., that there is a HL-causal place p for $H, L \in \mathscr{L}$ such that $H \not\rightsquigarrow L$. This means that $p \in {}^\bullet l \cap h^\bullet$ for transitions $h, l \in T$ such that $\lambda h = H$ and $\lambda l = L$ and there is a reachable marking $m \in [m_0\rangle$ satisfying $m[h\tau l\rangle$, $m[\tau\rangle$ and $\langle m[\tau\rangle(p) < F(p, l)$, with $\tau \in T^*_{\overline{\uparrow H}}$. If we let $U = \{H\}$, we can see that p is a causal place in the system $(N \setminus T_{\Uparrow U}, m)$ as well. In fact, $\Uparrow U = \uparrow H \setminus \{H\}$ and thus, since the firing sequence $\tau \in T^*_{\overline{\uparrow H}}$, it is not affected by the pruning. Again by by Theorem 1, we deduce that $\mathbf{N}$ is not U-BINI and thus not BINI, thus reaching a contradiction.

An analogous argument applies to the case in which $\mathbf{N}$ contains a conflict place. □



$\mathcal{D}$

Figure 8: Three-level security domain $\mathscr{D}$ with downgrading.

We next observe that, as in the transitive case, in order to detect security violations for BINI we do not need to examine all subsets of the security domain, but we can restrict to singletons, i.e. to subsets of the kind $U = \{L\}$.

Proposition 3 (multilevel BINI on single levels). *A net system* $\mathbf{N}$ *is BINI iff* $\mathbf{N}$ *is* $\{L\}$*-BINI for each* $L \in \mathscr{L}$*.*

Proof. The fact that if a net is BINI then it is $\{L\}$-BINI for each $L \in \mathscr{L}$ is a straight consequence of Definition 21. For the converse implication, observe that if $\mathbf{N}$ is not BINI then, by Definition 21, there is some $U \subseteq \mathscr{L}$ such that $\mathbf{N}$ is not U-BINI. By Lemma 6 this implies the existence of a HL-causal or conflict place for some levels $H, L \in \mathscr{L}$. From this fact, by the same lemma, we have that $\mathbf{N}$ is not $\{H\}$-BINI, as desired. □

Now we can prove that BINI in a multilevel domain can be reduced to BINI for three-levels systems. Let us denote by $\mathscr{D}$ the three-level domain in Fig. 8, where we have a *High* level, a *Low* level and a downgrading level that can mediate the flows from *High* to *Low*. The notion of BINI in a net system over $\mathscr{D}$, apart from a slightly different presentation, is exactly the one studied in [13].

Definition 22 (mapping to three-levels). *Let* $\mathbf{N} = ((P, T, \lambda, F), m_0)$ *be a net systems over a multilevel security domain* $\mathscr{L}$*. For each* $L \in \mathscr{L}$ *we define a net system over the three-level domain* $\mathscr{D}$ *as* $\Phi_L(\mathbf{N}) = ((P, T, \lambda', F), m_0)$ *where*

$$\lambda'(t) = \begin{cases} \mathit{High} & \text{if } t \in T_L \\ \mathit{Down} & \text{if } t \in T_{\Uparrow L} \\ \mathit{Low} & \text{if } t \in T_{\overline{\uparrow L}} \end{cases}$$

Proposition 4 (multilevel BINI to 3-level BINI). *A net system* $\mathbf{N}$ *is BINI if and only for each* $L \in \mathscr{L}$ *the three-levels system* $\Phi_L(\mathbf{N})$ *is BINI.*

Proof. By Proposition 3, we know that $\mathbf{N}$ is BINI iff it is $\{L\}$-BINI for any $L \in \mathscr{L}$. In order to conclude observe that $\mathbf{N}$ is $\{L\}$-BINI iff $\Phi_L(\mathbf{N})$ is BINI. By Definition 21, $\Phi_L(\mathbf{N})$ is BINI if it is U-BINI for every subset $U \subseteq \mathscr{D}$. It is easy to verify that this is a non-trivial requirement only for $U = \{\mathit{High}\}$. Since $\Uparrow\{\mathit{High}\} = \{\mathit{Down}\}$, applying Definition 20, we get that $\Phi_L(\mathbf{N})$ is $\{\mathit{High}\}$-BINI if the pruned net system $N \setminus T_{\{\mathit{Down}\}}$ is $\{\mathit{High}\}$-BNDC in the transitive domain $(\{\mathit{High}, \mathit{Low}\}, \rightsquigarrow^*_{\mathscr{D}})$. Recalling how the relabeling

has been defined, it is easy to see that $N \setminus T_{\{Down\}}$ is the relabeling of $N \setminus T_{\Uparrow L}$. Moreover requiring that $N \setminus T_{\{Down\}}$ is $\{High\}$-BNDC in the domain $(\{High, Low\}, \rightsquigarrow^*_{\mathscr{D}})$ is the same as requiring that $N \setminus T_{\Uparrow L}$ is $\{L\}$-BNDC in the domain $(\mathscr{L} \setminus \Uparrow L, \rightsquigarrow^*)$. Therefore Proposition 3 allows us to conclude. □

4.3. Characterizing Multilevel BINI in Safe Net Systems

In this section we show that focusing on safe net systems [7] it is possible to identify weaker notions of causal and conflict place, providing a characterization of BINI amenable of effective verification in the unfolding. While the notions of causal and conflict places were exactly the same in the transitive and intransitive case, their weak variants need to be slightly changed when moving to intransitive domains.

Definition 23 (intransitive weak causal place). *Let $H, L \in \mathscr{L}$ be security levels such that $H \not\rightsquigarrow L$. An* intransitive weak HL-causal place *in a net system $\mathbf{N}$ is a place $p \in {}^\bullet l \cap h^+$, for some $l, h \in T$ such that $\lambda h = H$, $\lambda l = L$, and there exists a reachable marking $m \in [m_0\rangle$ such that $m[h\tau l\rangle$, with $\tau \in T^*_{\overline{\Uparrow H}}$. Place p is called* intransitive weak causal *if it is a intransitive weak HL-causal for some $H, L \in \mathscr{L}$.*

Definition 24 (intransitive weak conflict place). *Let $H, L \in \mathscr{L}$ be two security levels such that $H \not\rightsquigarrow L$. An* intransitive weak HL-conflict place *in a net system $\mathbf{N}$ is a place $p \in {}^\bullet l \cap h^-$, for some $l, h \in t$ such that $\lambda h = H$, $\lambda l = L$, and there exists a reachable marking $m \in [m_0\rangle$ such that $m[h\rangle$ and $m[\tau l\rangle$, with $\tau \in t^*_{\overline{\Uparrow H}}$. Place p is called* intransitive weak conflict *if it is a intransitive weak HL-conflict for some $H, L \in \mathscr{L}$.*

The difference with respect to the notions of weak causal and conflict place given in Section 3.3 for transitive policies is that here we require τ not to contain any transition to which information can could legally flow from h, according to the policy. Intuitively, the reason is that, otherwise, the flow from h to l would be mediated by such transition, possibly making the flow legal.

Theorem 5 (BINI through intransitive causal and conflict places). *A safe net system $\mathbf{N}$ is BINI iff it contains no intransitive weak causal or conflict place.*

Proof. Assume that $\mathbf{N}$ is not BINI. Then there is $U \subseteq \mathscr{L}$ such that $\mathbf{N}$ is not U-BINI. In turn, this means that there exists a reachable marking $m \in [m_0\rangle$ for which $(\mathbf{N} \setminus T_{\Uparrow U}, m)$ is not U-BNDC in the transitive domain $\mathscr{L}' = (\mathscr{L} \setminus \Uparrow U, \rightsquigarrow^*)$. By Lemma 1 we know that this implies the existence of a HL-causal or conflict place in the pruned net $(\mathbf{N} \setminus T_{\Uparrow U}, m)$ in the transitive domain $\mathscr{L}'$, for levels $L, H \in \mathscr{L}$ such that $H \in U$ and $L \in \overline{U}$ (whence, in particular, $H \not\rightsquigarrow L$).

Assume, e.g., that there is a HL-causal place p. This means that $p \in {}^\bullet l \cap h^\bullet$ for some $l, h \in T$ such that $\lambda h = H$, $\lambda l = L$, and there exists a reachable marking $m \in [m_0\rangle$ such that $m[h\tau l\rangle$, $m[\tau\rangle$ and $\langle m[\tau\rangle(p) < F(p, l)$, with $\tau \in (T'_{\overline{\uparrow H}})^*$, where $T' = T \setminus T_{\Uparrow U}$ is the set of transitions of $(\mathbf{N} \setminus T_{\Uparrow U}, m)$. It follows that $\tau \in (T_{\overline{\Uparrow H}})^*$ in the original domain $\mathscr{L}$ and thus p is an intransitive weak HL-causal place in $\mathbf{N}$ with respect to the intransitive domain $\mathscr{L}$. If p were a HL-conflict place we can proceed analogously and conclude that p is a weak HL-conflict place also in the intransitive domain.

For the converse implication, assume that $\mathbf{N}$ contains an intransitive weak causal or conflict place p. Suppose, e.g., that p is an intransitive weak HL-causal place, namely $p \in {}^\bullet l \cap h^+$, for some $l, h \in T$ such that $H = \lambda h \not\rightsquigarrow \lambda l = L$, and there exists a reachable marking $m \in [m_0\rangle$ such that $m[h\tau l\rangle$, with $\tau \in (T_{\overline{\Uparrow H}})^*$. Now, take $U = \{H\}$ and consider the pruned system $(\mathbf{N} \setminus T_{\Uparrow U}, m)$. In such system we have exactly the same firing sequence showing that p is a weak causal place in the transitive domain $\mathscr{L}' = (\mathscr{L} \setminus \Uparrow U, \rightsquigarrow^*)$. The net system $(\mathbf{N} \setminus T_{\Uparrow U}, m)$ is safe since $\mathbf{N}$ is. Therefore by Theorem 2, the net $(\mathbf{N} \setminus T_{\Uparrow U}, m)$ is not U-BNDC and thus $\mathbf{N}$ is not BINI. □

Consider again the net system $\mathbf{S}$ in Fig. 1. In Section 3.2 we observed that $\mathbf{S}$ is not BNDC due to some interferences between the cache and the sensors, and between the sensors themselves. In both cases the interferences were stemming out from the mutually exclusive access to the cache. If this mode of access is a requisite of the system, or if it is an hardware constraint, it may well be the case that the developer will want to ignore those interferences, deeming them inevitable and not problematic. One way to do so could be changing the old transitive policy into an intransitive one and adding the flows $C \rightsquigarrow A$ and $C \rightsquigarrow B$. In this way, places a_0, b_0 and *Free* would no longer be causal, but place *Free* would still be conflict. Furthermore, altering the policy in this way is a poor modeling practice, because it allows all sorts of other interferences to occur between the cache and the sensors, whereas the goal was only to amend the inevitable and known ones.

A better solution is to introduce an appropriate number of additional "downgrading" levels to the security domain, and to modify the net by adding downgrading transitions that play the role of "explicit casts" of the interferences to be ignored. In Fig. 9 we show how this can be done in order to make the net system BINI (we only show a part of the system: the processor is unchanged and the second sensor is symmetric to the first one). In the coloured version, downgrading transitions $d1_D$, $d2_D$ and $d3_D$ are highlighted in green. As the reader can see, transition $d3_D$ is causally included between transitions $update1_C$ and transition get_A, and it downgrades the former to level D, so that place a_0 is no longer causal for the latter. In a slightly more complex way, transitions $d1_D$ and $d2_D$ achieve the same result for place *Free*.

4.4. Characterizing BINI in the Unfolding of Safe Nets

In this section we give a characterization of BINI for safe nets based on the unfolding semantics: as in the transitive case we reduce the satisfaction of the property BINI to a structural property of the unfolding of the net. This is done by showing how occurrences of intransitive weak causal or conflict places can be characterized in the unfolded net.

Theorem 6 (intransitive weak causal/conflict places in the unfolding). *Let* $\mathbf{N}$ *be a net system and let* p *be a place in* $\mathbf{N}$*. Then*

(i) p *is an intransitive weak causal place in* $\mathbf{N}$ *iff there exists a condition* b *in* $\mathcal{U}(\mathbf{N})$ *such that* $\pi_1(b) = p$ *and there are events* h', l' *such that*

1. $b \in {}^\bullet l' \cap h'^+$,
2. *for all events* e *such that* $h' < e' \leq l'$ *it holds* $\lambda h' \not\rightsquigarrow \lambda e$.

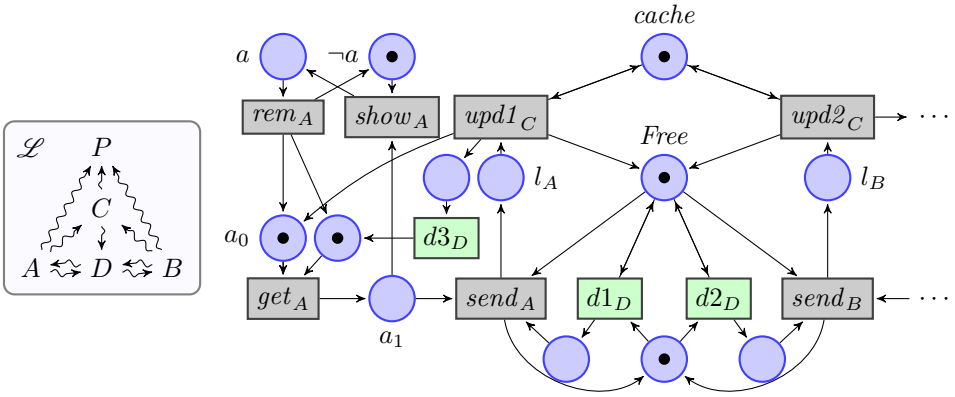


Figure 9: A possible fix of the net systems $\mathbf{S}$ that makes it BINI. Only part of the system is shown, and as usual $\lambda x_L = L$.

(ii) *p is an intransitive weak conflict place in $\mathbf{N}$ iff there exists a condition b in $\mathcal{U}(\mathbf{N})$ such that $\pi_1(b) = p$ and there are events h', l' such that*

1. *$b \in {}^\bullet l' \cap h'^-$,*
2. *$C = [h') \cup [l'] \in \mathcal{C}(\mathcal{U}(\mathbf{N}))$ and for all $d' \in C$ if $\lambda h' \rightsquigarrow \lambda d'$ and $d' \leq l'$ then $\neg(d' \# h')$.*

Proof. (i) Let p be an intransitive weak causal place in $\mathbf{N}$. Then $p \in {}^\bullet l \cap h^+$ for transitions l, h such that $\lambda h \not\rightsquigarrow \lambda l$, and there exist $m \in [m_0\rangle$ and $\tau \in T^*_{\nmid \lambda h}$ such that $m[h\tau l\rangle$. Therefore in the unfolding there are occurrences of p, l, h, namely a condition b and events l', h' such that $\pi_1(b) = p$, $\pi_1(h') = h$, $\pi_1(l') = l$ (hence $\lambda h' \not\rightsquigarrow \lambda l'$) and $b \in {}^\bullet l' \cap h'^+$. Furthermore for every e such that $h' < e \leq l'$, the corresponding transition $\pi_1(e)$ occurs in τl. Since $\tau \in T^*_{\nmid \lambda h}$ and $\lambda h' \not\rightsquigarrow \lambda l'$ we deduce that $\lambda h' \not\rightsquigarrow \lambda e$ as desired.

Vice versa, assume that $h = \pi_1(h')$, $l = \pi_1(l')$ and $p = \pi_1(b)$, for some condition b and events l', h' such that $b \in {}^\bullet l' \cap h'^+$, and furthermore for all e such that $h' < e \leq l'$ it holds $\lambda h \not\rightsquigarrow \lambda e$. First, observe that $p \in {}^\bullet l \cap h^+$ and $\lambda h \not\rightsquigarrow \lambda l$. Consider the set $C = \{e \in [l') \mid [e] \cap [h'] = \emptyset\}$, which is a configuration: it is conflict-free because $C \subseteq [l')$ and it is down-closed because if $e \in C$ and $e' \leq e$ then $e' \leq e < l'$, namely $e' \in [l')$. Moreover $[e'] \subseteq [e]$ and thus $[e'] \cap [h'] \subseteq [e] \cap [h'] = \emptyset$. Now consider the markings $m_1 = \mathsf{M}([h') \cup C)$ and $m_2 = \mathsf{M}([l'))$ and let τ be any linearization of $[l') \setminus ([h'] \cup C)$ compatible with causality. Notice that $([h'] \cup C) \subseteq [l')$ and hence is indeed a configuration, and $[l') \setminus ([h'] \cup C)$ is precisely the set of all e' such that $h' < e' < l'$. Then $m_1[h\pi_1(\tau)\rangle m_2[l\rangle$ and furthermore $\lambda\tau \subseteq \overline{\uparrow \lambda h'}$, hence p is an intransitive weak causal place in $\mathbf{N}$.

(ii) Let p be an intransitive weak conflict place in $\mathbf{N}$. Then $p \in {}^\bullet l \cap h^-$ with $\lambda h \not\rightsquigarrow \lambda l$, and there exists $m \in [m_0\rangle$ and $\tau \in T^*_{\nmid \lambda h}$ such that $m[h\rangle$ and $m[\tau l\rangle$. Therefore in the unfolding there are occurrences of p, l, h, namely a condition b and events l', h' such that $\pi_1(b) = p$, $\pi_1(h') = h$, $\pi_1(l') = l$, with $b \in {}^\bullet l' \cap h'^-$. Moreover there are configurations $C' \subseteq C$

such that $\mathsf{M}(C') = m$, $\mathsf{M}(C) = \langle m[\tau l\rangle$ and $C \supseteq [h') \cup [l']$. Hence $[h') \cup [l'] \in \mathcal{C}(\mathcal{U}(\mathbf{N}))$. Furthermore, also the second part of condition (2) holds: consider any d' such that $\lambda h' \rightsquigarrow \lambda d'$ and $d' < l'$, we reason by cases. If $d' \in C'$, then $\neg(d' \# h')$ because $\mathsf{M}(C')[h\rangle$. Otherwise, if $d' \in C \setminus C'$ then it means $d' \in \tau l$ and thus $\lambda h' \not\rightsquigarrow \lambda d'$ against the hypotheses.

Vice versa, assume that conditions (1) and (2) are satisfied. By (1), if we let $h = \pi_1(h')$, $l = \pi_1(l')$, we have that $p \in {}^\bullet l \cap h^-$ and $\lambda h \not\rightsquigarrow \lambda l$. We need to prove the existence of m and τ satisfying the remaining conditions in the definition of intransitive weak conflict place (Definition 24). Consider the set $D = \{d' \in [l') \mid \neg(d' \# h')\}$, we claim that $C = [h') \cup D$ is a configuration. To see that C is down-closed consider $d' \in C$ and $d'' < d'$. If $d' \in [h')$, then clearly $d'' \in [h') \subseteq C$. If $d' \in D$ then also $d'' \in D \subseteq C$. In fact, since $d' \in [l')$ then also $d'' \in [l')$. Moreover, since $\neg(d' \# h')$, $d'' < d'$ and conflict is inherited necessarily $\neg(d'' \# h')$. To see that C is conflict-free assume for the sake of contradiction $d', d'' \in C$ and $d' \# d''$. First of all, since both $[h')$ and $D \subseteq [l')$ are conflict-free, the only possibility is that $d' \in [h')$ and $d'' \in D$. But then again from $d' < h$ and the inheritance of $\#$ with respect to $<$ we would have $d'' \# h$, producing contradiction. Thus C is indeed a configuration, and furthermore $\mathsf{M}(C)[h\rangle$ by construction.

Call τ any linearization of $[l') \setminus C$ consistent with causality. Then $\mathsf{M}(C)[\tau l\rangle$ as well. In order to conclude that p is an intransitive weak conflict place we only need to prove that $\tau \in (T_{\overline{\uparrow \lambda h}})^* \supseteq (T_{\overline{\Uparrow \lambda h}})^*$. Suppose this is false. Then there is $t \in \tau$ such that $\lambda h \rightsquigarrow \lambda t$. But since $t = \pi_1(e)$ for some $e < l'$, we would have by hypothesis that $\neg(e \# h')$ and thus $e \in D$, a contradiction since τ does not contains elements of $C \supseteq D$. □

Checking condition (ii) is much harder than condition (i). The problem can be overcome since the construction of the causal reduct in Definition 16 still works as intended also in the intransitive case. This leads us to the following.

Proposition 5 (BINI in the causal reduct). *Let* $\mathbf{N}$ *be a net system. Then* $\mathbf{N}$ *is not BINI iff* $\gamma(\mathbf{N})$ *contains an intransitive causal place.*

Proof. Analogous to the proof of Proposition 1. □

4.5. Unfolding-based Algorithm for Multilevel BINI in Safe Net Systems

We next show how the characterization of BINI in the unfolding can be helpful for devising an algorithm for verifying the validity of such property in a safe net system. We try to adhere as much as possible to the structure of Subsection 3.5, highlighting the changes due to the use of intransitive policies.

4.5.1. Complete prefixes for multilevel BINI interferences.

In order to build a complete prefix, as in the transitive case, we need to enrich the marking associated with a configuration C with a function Λ_C, which records the security levels of the transitions that generated the tokens in each place. However, due to the intransitivity of the policy, this could be no longer sufficient to detect a violation. In fact, assume that an event l of level L consumes a token of level H such that $H \not\rightsquigarrow L$. Apparently this represents a violation of the policy since the presence of a token of level H reveals that an event of the same level, say h, have been executed before, making this fact visible at level L. However, this might not be a problem, since it could well be that a token of a level D such that $H \rightsquigarrow D \rightsquigarrow L$, is also in the pre-set of l, produced by an event

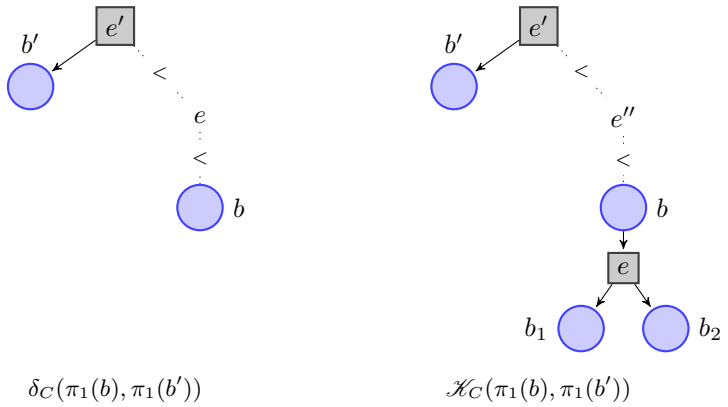


Figure 10: The definition of relations δ_C (left) and $\mathscr{K}_C$ (right), graphically.

d such that $h < d < l$. In this case, the flow of information from L to H is legitimately mediated by D. Clearly, a situation like this cannot happen within a transitive security domain, since by transitivity we could not have $H \rightsquigarrow D \rightsquigarrow L$ and $H \not\rightsquigarrow L$.

Roughly, in the situation above, we can think that the token of level D absorbs the token of level H to its level. In order to take care of this phenomenon we further enrich the markings with an *absorbing relation* δ_C over the places of the marking of a configuration.

Definition 25 (i-marking, i-complete prefix). *Let* $\mathbf{N}$ *be a safe net system. Given a configuration* $C \in \mathcal{C}(\mathcal{U}(\mathbf{N}))$*, the* intransitive confidentiality marking (i-marking) *of* C *is* $\mathsf{M}_i^*(C) = \langle \mathsf{M}(C), \Lambda_C, \delta_C \rangle$*, where* $\Lambda_C : P \to 2^{\mathscr{L}}$ *is the function defined as:*

$$\Lambda_C(p) = \{\lambda e \mid e \in T^{(\omega)} \ \wedge \ p \in \pi_1(e^+ \cap C^\circ)\}$$

and δ_C *is the binary relation on the marking* $\mathsf{M}(C)$*:*

$$\delta_C = \{(\pi_1(b), \pi_1(b')) \mid b, b' \in C^\circ \ \wedge \ \exists e, e' \in C \,.\, (b' \in e'^\bullet \ \wedge \ e' < e < b \ \wedge \ \lambda e' \rightsquigarrow \lambda e)\}.$$

A prefix $\mathcal{U}_F$ *of* $\mathcal{U}(\mathbf{N})$ *is* complete for i-marking reachability*, or simply* i-complete*, when for any configuration* $C \in \mathcal{C}(\mathcal{U}(\mathbf{N}))$ *there exists* $C' \in \mathcal{C}(\mathcal{U}_F)$ *such that* $\mathsf{M}_i^*(C) = \mathsf{M}_i^*(C')$*.*

As in the transitive case, Λ_C maps each place p in the marking $\mathsf{M}(C)$ to the set of security levels of the transitions that generated tokens in p and, since the net is safe, this is actually a singleton or the empty set. Concerning relation δ_C, intuitively, whenever $\delta(p, p')$ we know that that the token p can absorb the token p' to its level, in the sense explained above. The situation is schematized in Fig. 10 (left): there are $b, b' \in C^\circ$ such that $\pi_1(b) = p$, $\pi_1(b') = p'$, the event e' generating b' is a causal ancestor of b and there is e such that $e' < e \leq b$ such that $\lambda e' \rightsquigarrow \lambda e$. When b, b' are consumed by the same event, say e'', the flow from e' to e'' is necessarily mediated by e' through b. Thus we only need to check that b is of a legal level. Hence we can intuitively think that b' is absorbed to the level of b.

An i-complete prefix $\mathcal{U}_F$ of $\mathcal{U}(\mathbf{N})$ includes sufficient information for deciding whether or not $\mathbf{N}$ contains an intransitive weak causal place.

Theorem 7 (intransitive weak causal places in i-complete prefixes). *Let* $\mathbf{N}$ *be a safe net system and let* $\mathcal{U}_F$ *be an i-complete prefix of* $\mathcal{U}(\mathbf{N})$*. Then* p *is an intransitive weak causal place in* $\mathbf{N}$ *iff there are in* ${\mathcal{U}_F}^{\triangleright}$ *a condition* b *such that* $\pi_1(b) = p$*, and events* h'*,* l' *such that* $b \in {}^\bullet l' \cap h'^{+}$*,* $\lambda h' \not\rightsquigarrow \lambda l'$*, and for all* $b' \in {}^\bullet l'$ *it holds* $\neg\delta_{[l')}(\pi_1(b'), p)$*.*

Proof. ($\Rightarrow$) Suppose that p in $\mathbf{N}$ is an intransitive weak causal place. Then by Theorem 6(i) there exists a condition b in $\mathcal{U}(\mathbf{N})$ such that $\pi_1(b) = p$ and events h', l' such that $b \in {}^\bullet l' \cap h'^{+}$ and for all e such that $h' < e \leq l'$ we have $\lambda h' \not\rightsquigarrow \lambda e$. Consider the configuration $C = [l')$.

Observe that $\neg\delta_C(\pi_1(b'), p)$ for all $b' \in {}^\bullet l'$. In fact, if for some $b' \in {}^\bullet l'$ we have $\delta_C(\pi_1(b'), p)$, we would get the existence of $e \in C$ such that $\lambda h' \rightsquigarrow \lambda e$ and $h' < e \leq {}^\bullet b' < l'$, contradicting the hypotheses.

Now, from i-completeness of $\mathcal{U}_F$, there exists $C' \in \mathcal{C}(\mathcal{U}_F)$ such that $\mathsf{M}_i^*(C) = \mathsf{M}_i^*(C')$. From $\Lambda_C = \Lambda_{C'}$ and $\mathsf{M}(C) = \mathsf{M}(C')$, we deduce the existence of $b'' \in {C'}^{\circ}$ such that $\pi_1(b'') = \pi_1(b) = p$ and $\Lambda_{C'}(p) = \Lambda_C(p)$, and an event $l'' \in b''^{\bullet}$ such that ${}^\bullet l'' \subseteq {C'}^{\circ}$ (hence l'' in ${\mathcal{U}_F}^{\triangleright}$) with $\pi_1(l'') = \pi_1(l')$. From the fact that $\Lambda_C(p) = \Lambda_{C'}(p)$ we deduce the existence of $h'' \in C'$ such that $b'' \in h''^{+}$ and $\lambda h'' = \lambda h' = \Lambda_C(p)$. Summing up, we have $b'' \in h''^{+} \cap {}^\bullet l''$, $\lambda h'' = \lambda h' \not\rightsquigarrow \lambda l' = \lambda l''$ and $\pi_1(b'') = p$. Furthermore, from $\delta_C = \delta_{C'}$ we know that for all $b''' \in {}^\bullet l''$ it holds $\neg\delta_{C'}(\pi_1(b'''), p)$, as desired.

($\Leftarrow$) Suppose that there exists in ${\mathcal{U}_F}^{\triangleright}$ a condition b such that $\pi_1(b) = p$, and events h', l' such that $b \in {}^\bullet l' \cap h'^{+}$, $\lambda h' \not\rightsquigarrow \lambda l'$, and for all $b' \in {}^\bullet l'$ it holds $\neg\delta_{[l')}(\pi_1(b'), p)$. From the last condition we deduce that for all e such that $h' < e \leq l'$ we have that $\lambda h' \not\rightsquigarrow \lambda e$. Therefore we can use Theorem 6(i) to deduce that p is an intransitive weak causal place. □

Corollary 5 (BINI on i-complete prefixes). *Let* $\mathbf{N}$ *be a safe net system and let* $\mathcal{U}_F$ *be a i-complete prefix of* $\mathcal{U}(\gamma(\mathbf{N}))$*. Then* $\mathbf{N}$ *is not BINI iff* ${\mathcal{U}_F}^{\triangleright}$ *contains a condition* b *and events* h', l' *such that* $b \in {}^\bullet l' \cap h'^{+}$*,* $\lambda h' \not\rightsquigarrow \lambda l'$*, and* $\neg\delta_{[l')}(\pi_1(b'), p)$ *for all* $b' \in {}^\bullet l'$*.*

From Corollary 5 we deduce that given a safe net one can check for BINI on a i-complete prefix of the unfolding of its causal reduct. This point is developed in the next section.

4.5.2. Unfolding-based algorithm for multilevel BINI on safe net systems.

Despite the fact that i-markings are sufficient to characterize BINI, they do not carry enough information for an inductive construction of a complete prefix. This is explained with the aid of Fig. 11. Consider the configurations $C_1 = \{h^1, l^2, t^5\}$ and $C_2 = \{l^1, h^2, t^4\}$. They have the same i-marking $\mathsf{M}_i^*(C_1) = \mathsf{M}_i^*(C_2)$. In fact, $\mathsf{M}(C_1) = \mathsf{M}(C_2) = \{p, q, s\}$, and $\Lambda_{C_1} = \Lambda_{C_2} = \{p \mapsto \lambda h, q \mapsto \lambda l, s \mapsto \lambda t\}$, and $\delta_{C_1} = \delta_{C_2} = \emptyset$. Therefore, in principle, one of the two configurations could be discarded in the construction of the prefix. However, note that they enable an occurrence of transition d and, once extended with the corresponding event, their absorbing relations become different in that $\delta_{C_1}(p', p)$ while $\neg\delta_{C_2}(p', p)$. This is the reason why we need to further enrich the marking by recording what we call a kinfolk relation.

Definition 26 (ai-marking). *Let* $\mathbf{N}$ *be a net system and let* $C \in \mathcal{C}(\mathcal{U}(\mathbf{N}))$*. The* algorithmic intransitive confidentiality marking (ai-marking) *of* C *is* $\mathsf{M}_{ai}^*(C) = \langle \mathsf{M}_i^*(C), \mathscr{K}_C \rangle$*, where* $\mathscr{K}_C$ *is the binary relation over* $\mathsf{M}(C)$ *defined by:*

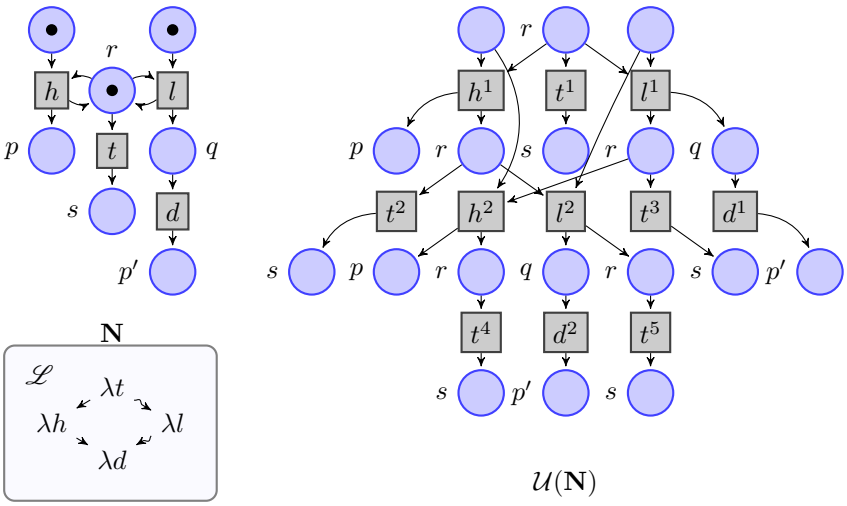


Figure 11: An example motivating the need of the kinfolk relation.

$$\mathscr{K}_C = \{(\pi_1(b), \pi_1(b')) \mid b, b' \in C^\circ \ \wedge \ \exists e' \in C \,.\, b' \in e'^\bullet \ \wedge \ e' < b\}$$

When $\mathscr{K}_C(p, p')$ we say that p is in the kinfolk of p'. The situation is schematized in Fig. 10 (right). As in the definition of δ_C, there are $b, b' \in C^\circ$ such that $\pi_1(b) = p$, $\pi_1(b') = p'$, the event e' generating b' is a causal ancestor of b, but here we do not require the existence of e such that $e' < e \leq b$ and $\lambda e' \rightsquigarrow \lambda e$. Hence, it might not be the case that $\delta_C(p, p')$. However, knowing that e' is an ancestor of b is relevant since if an event e such that $\lambda e \rightsquigarrow \lambda e'$ consumes b then all places in $e^\bullet$ (like those corresponding to b_1 and b_2 in the picture) will be in relation δ_C with p'.

Note that, with the aid of the kinfolk relation the configurations C_1 and C_2 in the example of Fig. 11 are taken apart. In fact, $\mathscr{K}_{C_1} = \{(q, p), (s, q)\}$ and $\mathscr{K}_{C_2} = \{(p, q), (s, q)\}$.

The notion of cut-off is thus updated by replacing the confidentiality marking with the algorithmic marking.

Definition 27 (algorithmic cut-off). *Let* $\mathbf{N}$ *be a net system and let* $\mathcal{U}_F$ *be a prefix of its unfolding* $\mathcal{U}(\mathbf{N})$*. An event* e *in* $\mathcal{U}_F$ *is called an* (algorithmic) cut-off *if there exists another event* e' *in* $\mathcal{U}_F$ *such that* $\mathsf{M}^*_{ai}([e]) = \mathsf{M}^*_{ai}([e'])$ *and* $[e'] \prec [e]$*.*

An algorithm for verifying whether a safe net system is BINI easily follows from the previous results. It is reported in Fig. 12. Note that with respect to the algorithm for checking BNDC in Fig. 6, we only need to update the test in the first `if` clause by adapting it to the condition from Theorem 7.

Corollary 6 (correctness of the algorithm for safe nets). *Let* $\mathbf{N}$ *be a safe net system. The algorithm of Fig. 12 always terminates and provides the answer 'yes' iff* $\mathbf{N}$ *is BINI.*

Proof. Observe that the equivalence of ai-markings over configurations is preserved by extension, namely given two configurations $C, C' \in \mathcal{C}(\mathcal{U}(\mathbf{N}))$, if $\mathsf{M}^*_{ai}(C) = \mathsf{M}^*_{ai}(C')$ and

Data: A safe net system $\mathbf{N}$.
compute $\gamma(\mathbf{N})$
$\mathcal{U}_F = \gamma(m_0)$
$pe = PE(\mathcal{U}_F)$
while $pe \neq \emptyset$ **do**
  take $e \in pe$ such that $[e]$ is $\prec$-minimal;
  if $\exists h, b \in \mathcal{U}_F.\ b \in {}^\bullet e \cap h^+ \ \wedge\ \lambda h \not\rightsquigarrow \lambda e \ \wedge\ \forall b' \in {}^\bullet e\,.\,\neg\delta_{[e)}(\pi_1(b'), \pi_1(b))$ **then**
    return *'no'*
  end
  add e to $\mathcal{U}_F$
  if *e is not an algorithmic cut-off* **then**
    add $e^\bullet$ to $\mathcal{U}_F$
  end
  $pe = PE(\mathcal{U}_F)$
end
return *'yes'*;

Figure 12: Algorithm to decide BINI on safe net systems.

$C \cup \{e\} \in \mathcal{C}(\mathcal{U}(\mathbf{N}))$ for some $e \notin C$ then there exists $e' \notin C'$ such that $C' \cup \{e'\} \in \mathcal{C}(\mathcal{U}(\mathbf{N}))$ and $\mathsf{M}^*_{ai}(C \cup \{e\}) = \mathsf{M}^*_{ai}(C' \cup \{e'\})$. It is easily seen that e' is the unique (since the net is safe) event enabled at C' such that $\pi_1(e) = \pi_1(e')$. Then by [19], the construction of a prefix based on ai-markings terminates producing a finite prefix of the unfolding complete for ai-markings. This is clearly also complete for i-markings and thus the desired result immediately follows from Corollary 5. □

5. Compositionality of BNDC and BINI

We conclude the theoretical part of the paper with the observation that BNDC and BINI are compositional with respect to the operations of (parallel) composition and restriction, as defined in Section 2.2. This can be helpful in the verification phase.

Let $\mathbf{N}$ and $\mathbf{N}'$ be net systems such that $\mathbf{N} \,|\, \mathbf{N}'$ is defined. Since the set of places of $\mathbf{N} \,|\, \mathbf{N}'$ is the (disjoint) union of the sets of places of $\mathbf{N}$ and $\mathbf{N}'$, given m and m' markings of $\mathbf{N}$ and $\mathbf{N}'$, respectively, clearly $m \,|\, m'$ defined as $m \cup m'$ is a marking of $\mathbf{N} \,|\, \mathbf{N}'$, and any marking of $\mathbf{N} \,|\, \mathbf{N}'$ is of such shape. In particular, the initial marking is $m_0 \,|\, m'_0$.

Given a firing sequence $m \,|\, m[\sigma\rangle$ we denote its projection to $\mathbf{N}$ as follows:

$$\sigma{\restriction_{\mathbf{N}}} = \begin{cases} \epsilon & \text{if } \sigma = \epsilon \\ \sigma'{\restriction_{\mathbf{N}}}\ t & \text{if } \sigma = \sigma' t \text{ and } t \in T \\ \sigma'{\restriction_{\mathbf{N}}} & \text{if } \sigma = \sigma' t \text{ and } t \notin T \end{cases}$$

The projection on $\mathbf{N}'$ is defined analogously.

Lemma 7 (soundness of projections). *Let $\mathbf{N}$ and $\mathbf{N}'$ be net systems such that $\mathbf{N} \,|\, \mathbf{N}'$ is defined. If $m \,|\, m'[\sigma\rangle m_1 \,|\, m'_1$ then $m[\sigma{\restriction_{\mathbf{N}}}\rangle m_1$ and $m'[\sigma{\restriction_{\mathbf{N}'}}\rangle m'_1$.*

PROOF. We only prove the statement for the projection on $\mathbf{N}$ (the one for $\mathbf{N}'$ is analogous). We proceed by induction on $|\sigma|$, the length of σ. If $|\sigma| = 0$ then $\sigma = \epsilon$ and the statement is trivial. Otherwise $\sigma = \sigma' t$, with $m \,|\, m'[\sigma'\rangle m_2 \,|\, m_2'[t\rangle m_1 \,|\, m_1'$.

By inductive hypothesis applied to $m \mid m'[\sigma'\rangle m_2 \mid m_2'$, we obtain $m[\sigma'\restriction_{\mathbf{N}}\rangle m_2$. We conclude by distinguishing various cases according to the nature of transition t. If $t \notin T$ then $\sigma' t\restriction_{\mathbf{N}} = \sigma'\restriction_{\mathbf{N}}$ and $m_1 = m_2$, hence $m[\sigma\restriction_{\mathbf{N}}\rangle m_1$, trivially. Otherwise, $\sigma\restriction_{\mathbf{N}} = \sigma'\restriction_{\mathbf{N}} t$. Moreover, the observation that the pre- and post-set of transition t in $\mathbf{N} \mid \mathbf{N}'$, when restricted to S, are exactly the original pre- and post-set of t in $\mathbf{N}$ allows us to conclude that $m_2[t\rangle m_1$. Thus $m[\sigma'\restriction_{\mathbf{N}} t\rangle m_1$, as desired. □

Proposition 6 (Compositionality of BNDC and BINI). *If two nets systems* $\mathbf{N}$ *and* $\mathbf{N}'$ *are BNDC (or BINI) then* $\mathbf{N} \,|\, \mathbf{N}'$ *is also BNDC (or BINI). Moreover, for any* $T' \subseteq T$*, then* $\mathbf{N} \setminus T'$ *is BNDC (or BINI).*

PROOF. Assume for the sake of contradiction that $\mathbf{N} \,|\, \mathbf{N}'$ is not BNDC. Then by Theorem 1 we know that it contains either a HL-causal or a HL-conflict place p for some $H, L \in \mathscr{L}$. We can assume without loss of generality $p \in P$.

If p is HL-causal then, by Definition 12, $p \in {}^\bullet l \cap h^\bullet$ for suitable transitions $h, l \in T \cup T'$ such that $\lambda h = H$ and $\lambda l = L$ and there exists a reachable marking $m \,|\, m' \in [m_0 \,|\, m_0'\rangle$ satisfying $m \,|\, m'[h\tau l\rangle$, $m \,|\, m'[\tau\rangle$ and $\langle m \,|\, m'[\tau\rangle(p) < F(p, l)$, with $\tau \in T^*_{\restriction H}$. From $p \in P$ we deduce also l, $h \in T$. Let σ be a firing sequence such that $m_0 \,|\, m_0'[\sigma\rangle m \,|\, m'$. By Lemma 7 applied to $\sigma\tau$ we can deduce that all the conditions hold in $\mathbf{N}$ once we substitute $\tau\restriction_{\mathbf{N}}$ for τ. Hence p is a causal place in $\mathbf{N}$, leading to a contradiction. The very same reasoning allows us to deduce that if p is conflict in $\mathbf{N} \,|\, \mathbf{N}'$, it is conflict also in $\mathbf{N}$, so $\mathbf{N} \,|\, \mathbf{N}'$ must be BNDC.

Concerning restriction, assume that $\mathbf{N} \setminus T'$ is not BNDC. Then there must exists a causal or conflict place p in $\mathbf{N} \setminus T'$. Since $\mathbf{N} \setminus T'$ has the same places as $\mathbf{N}$ and a subset of its transitions, it is immediate to see that p is a causal place also in $\mathbf{N}$.

The proof for BINI is analogous. Since the notions of transitive causal and conflict places are formally the same in the transitive and intransitive case we just need to use Theorem 4 instead of Theorem 1. □

Consider, for instance, the running example net system $\mathbf{S}$ as modified in Fig. 9. It is easy to see that the decomposition proposed in Fig. 2 can be adapted to the modified version. Then it is possible to verify separately that the two parallel components are BINI and thus conclude that also the full system is.

6. The tool MultiUBIC

The unfolding-based algorithms outlined in the previous sections are implemented in a tool called MultiUBIC [14]. It extends a previous tool UBIC which was limited to two-level security domains (possibly with downgrading). MultiUBIC inputs a security policy (transitive or intransitive) and a safe net system, and it checks whether the system satisfies BNDC (for transitive policies) or BINI (for intransitive policies). The security policy is specified in a format illustrated in Listing 1. It first specifies the type of the policy (transitive or intransitive) which determines the property, BNDC or BINI, respectively, to be checked, the number n of security levels the domain and a name for

Listing 1: Format for security policies

```
MSD
TRANSITIVE/INTRANSITIVE
LVL n
0 name
1 name
...
n−1 name
POLICY
i TO j
```

each level. Then the security policy is given in the form of a collection of clauses i TO j. If the policy is declared to be transitive, MultiUBIC will automatically compute the transitive closure of its clauses, hence the user just needs to specify direct flows. The net system have to be specified in the PEP's *ll_net* format [23], with one additional constraint: transition names must be of the form *name_level*, where *level* is one of the level names specified in the policy file.

A number of options can be specified as well, most notably whether only the first or all interferences must be found during the analysis.

Compared to "interleaving competitors", like the Petri Net Security Checker [24] and ANICA [25] (Automated Non-Interference Check Assistant, written in C++), based on the work [5], MultiUBIC was expected to inherit the good performances of its ancestor UBIC: the fact that it relies on a partial order semantics should lead to a gain of efficiency especially for highly concurrent systems, where the state explosion problem becomes more serious. Indeed, we experimented MultiUBIC on some batteries of tests that were already used for UBIC in [7]. These are two-level net systems implementing various kind of mutual exclusion protocols with a parametric number of parallel participants. In average, UBIC and MultiUBIC have similar execution times. Thus, as it was the case for UBIC, also MultiUBIC outperforms ANICA when the level of concurrency grows (the tests are included in MultiUBIC distribution [14]).

Given the above, we devoted some effort to a comparison of the relative performances of MultiUBIC and UBIC. This is particularly of interest since we showed that the verification of multi-level security policies can be also reduced to a number of checks in a two-level setting (with a third downgrading level, in the intransitive case). In the transitive case, a reduction based on a direct application of the definition of BNDC would be unacceptably expensive, since we should consider a two-level problem for each possible upper set in the security domain, a fact that possibly leads to a number of subproblems exponential in the number of levels. Actually, we have shown that we can limit to a linear number of two-level checks, one for each level (see Corollary 1). Similar considerations apply to BINI in the intransitive case, where a quantification over all possible subsets of levels can be replaced by a quantification over all levels (see Proposition 4). MultiUBIC, besides allowing for a direct solution of the multilevel problem comes equipped with facilities for performing such reduction.

Some experiments reveal that solving directly the original multi-level problem, typi-

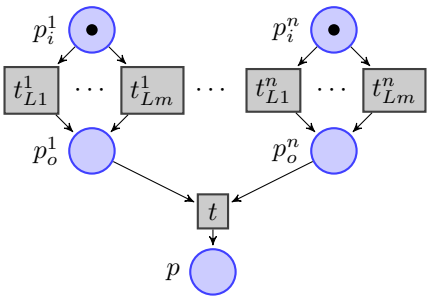


Figure 13: A net system where MultiUBIC behaves worse than UBIC.

cally provides a linear gain of efficiency, possibly at the price of an increase of memory usage. The performances of MultiUBIC can degrade for net systems where a relevant number of places have input transitions of different levels. This situation alone can, in a worst case scenario, make the direct solution of the multilevel problem perform definitively worse than the solution based on reduction, even when the number of levels increases.

For instance, consider the net system in Fig. 13, with a security domain $\mathscr{L} = \{L_1, \ldots, L_m\}$, and a trivial empty policy. The system consists of n blocks of m transitions in conflict, each of a different level, and all producing a token in a shared outgoing place p_o^j of the block. All such places are in turn consumed by a transition t, with an arbitrary security level in $\mathscr{L}$. The complete prefix of the unfolding built by MultiUBIC would include mn transitions for the blocks, plus m^n instances of the final transition t, one for each different enriched marking, hence $mn + m^n$ in total. The reductions method would instead produce m prefixes, each comprising mn transitions for the blocks, but only with 2^n instances of the final transitions, hence $m^2n + m2^n$ in total. If one fixes the number of levels (m is a constant) and analyze the complexity with respect to n, they both are exponential but, as soon as $m > 2$, the exponential of MultiUBIC is worse (since the basis is larger). More importantly, even if ones fixes the number of blocks (n is a constant) and increases the number of levels, as soon as $n > 2$, the complexity MultiUBIC $\Theta(m^n)$ becomes polynomially worse than that of UBIC, which is $\Theta(m^2)$. Therefore, in this example, MultiUBIC would need to generate a number of transitions larger then those produced with the reductions method, polynomially in m and exponentially in n. Even under the reasonable assumption that n is much larger than m this causes a sensible loss of performance. Several concrete experiments are presented in Appendix B, where they are used for a detailed comparison of the direct and reduction based approaches.

We conclude this section by remarking that MultiUBIC is a promising prototype but we think that there is room for improving its performances and usability with several enhancements:

- The major overhead of MultiUBIC lies in having to store data structures for relations δ and $\mathscr{K}$ as needed to deal with algorithmic intransitive markings. The construction of the causal reduct can heavily influence this overhead. In fact, it adds several self-looping transitions, the copies c_h of each transition h possibly involved in a conflict inference (see Definition 16) and in the unfolding this determines

the duplication of the whole branch of the starting with the execution of h. Fine grained optimizations of the construction are possible that, at the price of a more complicated definition, could limit the construction of these duplicated branches to the absolutely necessary ones, thus giving a sensible gain in performances.

- Along the same line, the places of the net system can be enriched with static annotations that enable the verifier to discard useless information and, correspondingly, to cut irrelevant branches of the unfolding. For instance, an immediate optimization consists in avoiding to record the level of the places that can be statically recognized as non-problematic, e.g. places $p \in P$ such that $\lambda t \rightsquigarrow \lambda t'$ for all $t \in {}^\bullet p$ and $t' \in p^\bullet$.

- At least in the transitive case, it seems possible to weaken cut-off condition, thus obtaining smaller prefixes. For example, if two histories produces the same enriched marking, but for the fact that some token in the second history has a higher confidentiality level, then the second history can be classified as a cut-off and discarded.

- Since in general MultiUBIC performs extremely better then UBIC with respect to the CPU time required by the verification, and the problematic cases are linked to a blow of memory consumption, MultiUBIC could benefit from a more CPU intensive approach.

7. Conclusions

We studied non-interference properties in a multilevel setting, both for transitive and intransitive security domains, focusing on Petri nets. Generalizing the work in [7, 13], we showed that Bisimilarity-based Non-Deducibility on Composition (BNDC) and its intransitive extension BINI [6], admit a causal characterizations in the unfolding semantics of safe net systems. This leads verification algorithms for BNDC and BINI on safe net systems with multilevel policies, implemented in the tool MultiUBIC.

The use of causality for deducing the occurrence of non-observable transitions from the occurrence of observable ones has been studied in [26], in the context of asynchronous diagnosis of discrete event event systems. The relation between diagnosability properties and non-interference is to be deepened, despite the fact that the former are trace-based while the latter is bisimulation-based. In this respect, the work on intransitive non-interference in [27], resorting to automata models and language theory for verification could be of help.

In the setting of Petri nets, other classes of information flow properties have been studied, like opacity properties [28] (which include non-interference) and selective non-interference [29]. Exploring the possibility of exploiting causal semantics in this general setting appears as an interesting and challenging venue of future research.

It would also be quite interesting to explore causal characterizations of non-interference for formalisms different from Petri nets, including process calculi and imperative languages, possibly through encodings of these formalisms into Petri nets. This would allow to establish a formal and possibly fruitful link between our work with the huge literature on non-interference in these settings (see, e.g., [2] and [30] for surveys). To this extent we are considering how the current theory would carry over classes of nets with additional features that could ease the encoding. Examples of features we would be able to include

are: read arcs (to improve concurrency, model readings of resources in a more efficient way, and mitigate the negative impact on performances of self loops), unboundedness, colored tokens (both of the latter to help cope with the encoding values).

We also plan to consider formalizations of non-interference obtained from the classical ones, by replacing interleaving observational semantics with true-concurrent ones [31]. The higher distinguishing power of such semantics could allow to identify new forms of interference which cannot be captured in an interleaving setting. Interesting reflections in this directions are reported in [32].

We are currently trying to use MultiUBIC to verify explicit information flows on choreography-based languages [33]. In the specific scenario we need to tackle, each entity of the choreography (variable, channel or principal) needs to be considered as belonging to a distinct security level, hence justifying the use of MultiUBIC.

[1] J. A. Goguen, J. Meseguer, Security policies and security models, in: Proceedings of the Symposium on Security and Privacy, IEEE Computer Society, 1982, pp. 11–20.

[2] R. Focardi, R. Gorrieri, Classification of security properties (Part I: Information flow), in: Proceedings of FOSAD'00, Springer, 2001, pp. 331–396.

[3] P. Ryan, Y. Schneider, Process algebra and non-interference, Journal of Computer Security 9 (1/2) (2001) 75–103.

[4] H. Mantel, Possibilistic definitions of security - an assembly kit, in: Proceedings of CSFW'00, IEEE Computer Society, 2000, pp. 185–199.

[5] N. Busi, R. Gorrieri, Structural non-interference in elementary and trace nets, Mathematical Structures in Computer Science 19 (6) (2009) 1065–1090.

[6] E. Best, P. Darondeau, R. Gorrieri, On the decidability of non interference over unbounded Petri nets, in: K. Chatzikokolakis, V. Cortier (Eds.), Proceedings of SecCo'10, Vol. 51 of EPTCS, Open Publishing Association, 2010, pp. 16–33.

[7] P. Baldan, A. Carraro, A causal view on non-intereference, Fundamenta Informaticae 140 (1) (2015) 1–38.

[8] D. McCullough, Noninterference and the composability of security properties, in: Symposium on Security and Privacy, IEEE Computer Society, 1988, pp. 178–186.

[9] J. Wittbold, D. Johnson, Information flow in nondeterministic systems, in: Symposium on Security and Privacy, IEEE Computer Society, 1990, pp. 148–161.

[10] M. Nielsen, G. Plotkin, G. Winskel, Petri nets, event structures and domains, part 1, Theoretical Computer Science 13 (1981) 85–108.

[11] D. E. Denning, A lattice model of secure information flow, Communication of the ACM 19 (5) (1976) 236–243.

[12] J. M. Rushby, Design and verification of secure systems, in: Proceedings of SOSP'81, ACM, 1981, pp. 12–21.

[13] P. Baldan, F. Burato, A. Carraro, Intransitive non-interference by unfolding, in: I. Lanese, E. Madelaine (Eds.), Proceedings of FACS'14, Vol. 8997 of LNCS, Springer, 2014, pp. 269–287.

[14] A. Beggiato, MultiUBIC, `https://github.com/AlessandroBeggiato/MultiUbic/releases`. `doi:10.5281/zenodo.242997`.

[15] Service Technology, ANICA: Automated Non-Interference Check Assistant, `http://service-technology.org/anica`.

[16] R. Gorrieri, M. Vernali, On intransitive non-interference in some models of concurrency, in: A. Aldini, R. Gorrieri (Eds.), Proceedings of FOSAD'11, Vol. 6858 of LNCS, Springer, 2011, pp. 125–151.

[17] J. Esparza, K. Heljanko, Unfoldings - A Partial order Approach to Model Checking, EACTS Monographs in Theoretical Computer Science, Springer, 2008.

[18] K. L. McMillan, A technique of state space search based on unfolding, Form. Methods Syst. Des. 6 (1) (1995) 45–65.

[19] V. Khomenko, M. Koutny, W. Vogler, Canonical prefixes of Petri net unfoldings, Acta Informatica 40 (2003) 95–118.

[20] J. Meseguer, U. Montanari, V. Sassone, Representation theorems for Petri nets, in: C. Freksa, M. Jantzen, R. Valk (Eds.), Foundations of Computer Science: Potential - Theory - Cognition, Vol. 1337 of LNCS, Springer, 1997, pp. 239–249.

[21] J. Rushby, Noninterference, transitivity, and channel-control security policies, Tech. rep. (Dec 1992). URL http://www.csl.sri.com/papers/csl-92-2/
[22] J. Esparza, S. Römer, W. Vogler, An improvement of McMillan's unfolding algorithm, Formal Methods in System Design 20 (20) (2002) 285–310.
[23] E. Best, B. Grahlmann, PEP Documentation and User Guide 1.8 (1998).
[24] S. Frau, R. Gorrieri, C. Ferigato, Petri net security checker: Structural non-interference at work, in: P. Degano, F. Guttman, J. Martinelli (Eds.), Proceedings of FAST'08, Vol. 5491 of LNCS, Springer, 2008, pp. 210–225.
[25] R. Accorsi, A. Lehmann, Automatic information flow analysis of business process models, in: A. Barros, A. Gal, E. Kindler (Eds.), Proceedings of BPM'12, Vol. 7481 of LNCS, Springer, 2012, pp. 172–187.
[26] S. Haar, Types of asynchronous diagnosability and the reveals-relation in occurrence nets, IEEE Transactions on Automatic Control 55 (10) (2010) 2310–2320.
[27] B. N. Hadj-Alouane, S. Lafrance, F. Lin, J. Mullins, M. M. Yeddes, On the verification of intransitive noninterference in multilevel security, IEEE Transactions on Systems, Man, and Cybernetics, Part B 35 (5) (2005) 948–958. doi:10.1109/TSMCB.2005.847749.
URL http://dx.doi.org/10.1109/TSMCB.2005.847749
[28] J. Bryans, M. Koutny, P. Ryan, Modelling dynamic opacity using Petri nets with silent actions, in: T. Dimitrakos, F. Martinelli (Eds.), Proceedings of FAST'05, Vol. 173 of LNCS, Springer, 2005, pp. 159–172.
[29] E. Best, P. Darondeau, Deciding selective declassification of Petri nets, in: POST'12, Vol. 7215 of LNCS, Springer, 2012, pp. 290–308. doi:10.1007/978-3-642-28641-4_16.
URL http://dx.doi.org/10.1007/978-3-642-28641-4_16
[30] H. Mantel, D. Sands, Controlled declassification based on intransitive noninterference, in: APLAS'04, 2004, pp. 129–145.
[31] R. van Glabbeek, U. Goltz, Refinement of actions and equivalence notions for concurrent systems, Acta Informatica 37 (4/5) (2001) 229–327.
[32] S. Fröschle, Causality, behavioural equivalences, and the security of cyberphysical systems, in: R. Meyer, A. Platzer, H. Wehrheim (Eds.), Correct System Design, Vol. 9360 of LNCS, Springer, 2015, pp. 83–98. doi:10.1007/978-3-319-23506-6_8.
URL http://dx.doi.org/10.1007/978-3-319-23506-6_8
[33] A. Lluch-Lafuente, F. Nielson, H. R. Nielson, Discretionary information flow control for interaction-oriented specifications, in: Logic, Rewriting, and Concurrency - Essays dedicated to José Meseguer on the Occasion of His 65th Birthday, Vol. 9200 of LNCS, Springer, 2015, pp. 427–450.
[34] M. Hack, Decidability Questions for Petri Nets, Outstanding Dissertations in the Computer Sciences, Garland Publishing, 1975.

Appendix A. Locally-safe nets

The results and algorithms that in the paper we developed for safe nets can be easily generalized to deal with a more general class of nets, named *locally-safe* net systems [7] that properly includes safe net systems. We next briefly point out the of adjustments of the theory needed for such a generalization.

Firstly, the notion of locally-safe net system itself have to be shifted to a multilevel setting.

Notation 4. *Given a net system* $\mathbf{N}$*, a place* $p \in P$ *and a transition* $t \in T$*, we set* $\mathsf{d}_t(p) = F(t,p) - F(p,t)$*, namely* $\mathsf{d}_t(p)$ *is the variation in the number of tokens in place* p *determined by the firing of* t*. Moreover we denote by* $t^- = \{p \in P : \mathsf{d}_t(p) < 0\}$ *and* $t^+ = \{p \in P : \mathsf{d}_t(p) > 0\}$ *the sets of places where the firing of* t *decreases and increases, respectively, the number of tokens.*

Definition 28 (locally-safe net system). *A net system* $\mathbf{N}$ *is* locally-safe *when for every* l, h *such that* $\lambda h \not\rightarrow \lambda l$*, and every* $m \in [m_0\rangle$ *:*

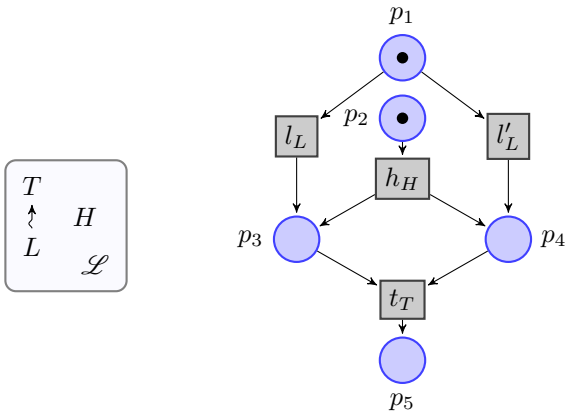


Figure A.14: A net system which is locally-safe but not safe.

- *if* ${}^\bullet l \cap h^+ \neq \emptyset$, *then* $\{p \in {}^\bullet l \cap h^+ : m(p) - \mathsf{d}_h(p) < F(p,l)\} \neq \emptyset$;
- *if* ${}^\bullet l \cap h^- \neq \emptyset$, *then* $\{p \in {}^\bullet l \cap h^- : m(p) + \mathsf{d}_h(p) < F(p,l)\} \neq \emptyset$.

In words, when ${}^\bullet l \cap h^+ \neq \emptyset$, i.e., transition h can generate tokens in a place in the pre-set of transition l, it is always the case that some token generated by h is essential to enable l. More precisely, in any reachable marking m there is at least one place $p \in {}^\bullet l \cap h^+$ such that $m(p) - \mathsf{d}_h(p) < F(p,l)$. Similarly, whenever ${}^\bullet l \cap h^- \neq \emptyset$, i.e., transition h can remove tokens in a place in the pre-set of transition l, we require that in any reachable marking m there is at least one place $p \in {}^\bullet l \cap h^-$ where the occurrence of h removes from p a number of tokens sufficient to disable l.

An example of net system which is locally-safe but not safe can be found in Fig. A.14, which also reports the associated domain $\mathscr{L}$. The net is clearly not safe since, for instance, after firing l and h a marking with two tokens in p_3 is reached. However, the net is locally-safe. In fact, the only potentially problematic situation concerns transitions h and t , since $H \not\rightsquigarrow T$ and $h^+ \cap {}^\bullet t = \{p_3, p_4\}$. Observe that any reachable marking m either $m(p_3) \leq 1$ or $m(p_4) \leq 1$. Hence, since $\mathsf{d}_h(p_3) = \mathsf{d}_h(p_4) = 1 = F(p_3,t) = F(p_4,t)$ either $m(p_3) - \mathsf{d}_h(p_3) < F(p_3,t)$ or $m(p_4) - \mathsf{d}_h(p_4) < F(p_4,t)$. In words, whenever t fires, it is always using in an essential way at least a token produced by h.

By a straightforward adaptation of the argument in [7], it is possible to show that local-safety can be reduced to coverability of a given place, a problem which is known to be decidable [34]. It is also immediate to see that the class of locally-safe net systems includes all safe net systems.

With the above definition, it is not difficult that the characterization of BNDC in Sections 3.3 and 4.3, and those for BINI in Sections 3.4 and 4.4, smoothly extend to locally-safe nets, just by replacing safe by locally-safe.

Also the algorithms for checking BNDC and BINI, respectively, in Sections 3.5 and 4.5 can be easily adapted. Only notice that in Definition 17 the function Λ_C has a proper set as value, rather than a singleton or empty set, since a place can contain more than one token. The same apply to Definition 25 where, in addition, the fact that there can

be multiple tokens in a place leads to define relation δ_C directly on the conditions of the prefix rather than on the places of the original net

$$\delta_C = \{(b, b') \mid b, b' \in C^\circ \ \wedge \ \exists e, e' \in C \,.\, (b' \in e'^\bullet \ \wedge \ e' < e < b \ \wedge \ \lambda e' \rightsquigarrow \lambda e)\}$$

Then we say that a prefix U of $\mathcal{U}(\mathbf{N})$ us i-complete when for any configuration $C \in \mathcal{C}(\mathcal{U}(\mathbf{N}))$ there exists $C' \in \mathcal{C}(U)$ such that $\mathsf{M}(C) = \mathsf{M}(C')$, $\Lambda_C = \Lambda_{C'}$ and there is a bijection between the cuts $\iota : C^\circ \to C'^\circ$ such that for all $b, b' \in C^\circ$ it holds $\delta_C(b, b')$ iff $\delta_{C'}(\iota(b), \iota(b'))$. Analogous modification are required for relation $\mathscr{K}_C$ in Definition 26, and the corresponding notion of completeness. Finally, note that here, since locally-safe nets are possibly unbounded, in order to ensure that a complete prefix is finite (whence termination of the algorithms) a boundedness hypothesis is needed.

Appendix B. Experimental results

In order to empirically compare the direct solution of a multilevel problem based on MultiUBIC with the approach based on reduction to two- (three-) level problems, we ran a battery of tests. More precisely, we compared MultiUBIC with a procedure whose behavior was:

1. Read a net system $\mathbf{N}$ with its multilevel security domain $\mathscr{L}$;
2. For each level $L \in \mathscr{L}$, produce a two- (or three-) level system $\mathbf{N}_L$;
3. Verify each $\mathbf{N}_L$ with UBIC;
4. Declare $\mathbf{N}$ secure in $\mathscr{L}$ if and only if UBIC declares secure every $\mathbf{N}_L$.

The verification was carried over a set of test nets $\mathbf{T}(n, l, e)$, designed to be parametric with respect to some characteristics that impact on the efficiency of the verification, in particular, the number of potential causal interferences that determines the size of the data structures (ai-markings) needed for the construction of the finite prefix, and the number potential conflict interferences, that influences the size of the causal reduct.

Figure B.15 helps in explaining how the test systems are built. Each system is a sequence of n blocks, where block i ($i \in \{1, \ldots, n\}$) consists of three conflicting transitions (t_{L_i}, $t'_{L'_i}$ and $t''_{L''_i}$) competing for common input place p_{i-1} and producing on an output place p_i, which in turn acts as input for the next block. The parameter l determine the number of different levels for transitions in each block. When $l = 1$ all levels coincide, i.e., $L_i = L'_i = L''_i$, when $l = 2$, we have $L_i = L'_i \neq L''_i$ and when $l = 3$ the levels are pairwise different. In words, l determines what we will refer as level clashes, i.e., the number of possible distinct levels for the token in the output place of the block. Especially in intransitive policies, where the history of a token is also relevant, this heavily influences the number of enriched markings.

Each level in a block is allowed to influence the corresponding one in the next block, i.e., $L_i \rightsquigarrow L_{i+1}$, $L'_i \rightsquigarrow L'_{i+1}$ and $L''_i \rightsquigarrow L''_{i+1}$. Other twisted "vertical" flows (e.g., from L_i to L'_{i+1}) are illegal. The legal interactions between levels inside a block are determined by parameter e. More precisely, the flow from each of the three levels L_i, L'_i and L''_i to another one in a block can be either allowed or not. Hence there are six potential flows and the parameter $e \in \{1, 2, \ldots, 6\}$ establishes how many of them are legal. This

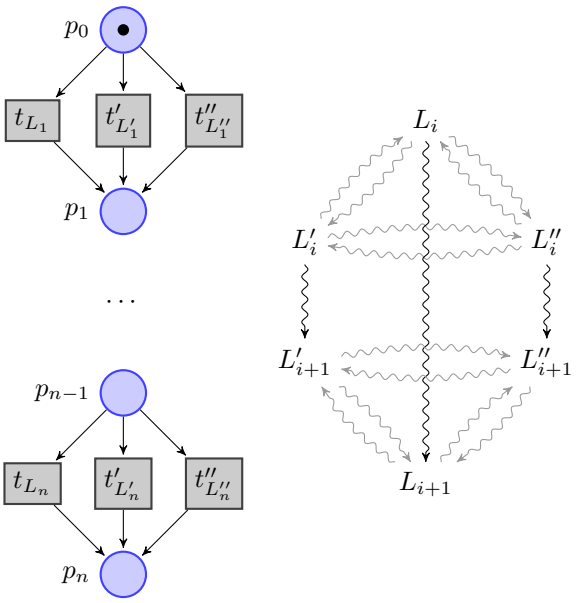


Figure B.15: A family of net systems used to test the tool.

influences the number of conflict places in the system. For example if $l = 3$ and $e = 6$ then there are no conflict places, while if $l = 3$ and $e = 0$ there are n conflict places. In this way we control how many transitions are added by the causal reduction.

We tested the systems against both transitive and intransitive policies. The data collected for both methods were: total time elapsed, memory usage, number of histories added to the prefix. Both UBIC and MultiUBIC were asked to produce a complete prefix, thus determining all possible interferences. Figure B.16 reports the ratio between the elapsed times of MultiUBIC and UBIC versus the cardinality of the MSD (which is in turn proportional to the size of the net system). Note that MultiUBIC overwhelmingly outperforms UBIC in terms of CPU time in most cases. It is noticeable that with the highest number of levels ($l = 3$), MultiUBIC gain starts to slightly decrease.

The situation changes in Fig. B.17, where we observe a reverse trend due to a combination of factors. As theoretically foresaw, an elevated number of level clashes sensibly diminish the gain in performances of MultiUBIC, to the point that it actually decreases with the growth of the cardinality of the security domain. But the critical factor seems to be the number of conflict places. Mapping onto three-levels system reduces the number of conflict places, thus reducing the overhead determined by the causal reduction. Because of the way we defined the causal reduction, if an intransitive policy is used, the number of different algorithmic markings greatly increases with the number of conflict places, and this overhead disrupt the advantage that MultiUBIC has over the reduction methods. This hypothesis is supported by Fig. B.18, where we directly plot the CPU time used by the two methods, for transitive and intransitive policies at the top and bottom respectively.

Finally, and to further support our claim, in Fig. B.19 we can see how UBIC has

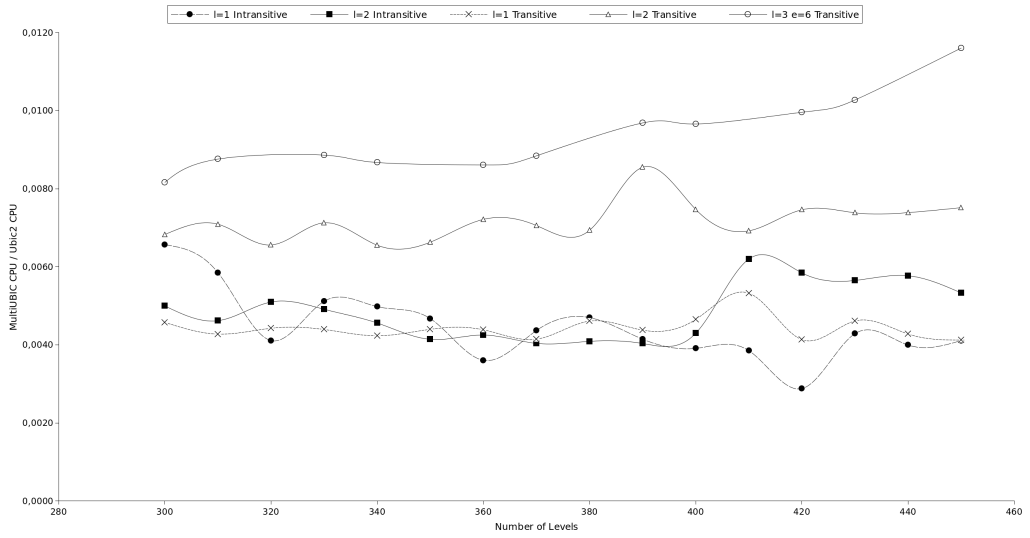


Figure B.16: Experimental comparison of MultiUBIC and UBIC: extremely high gain in CPU time achieved by MultiUBIC.

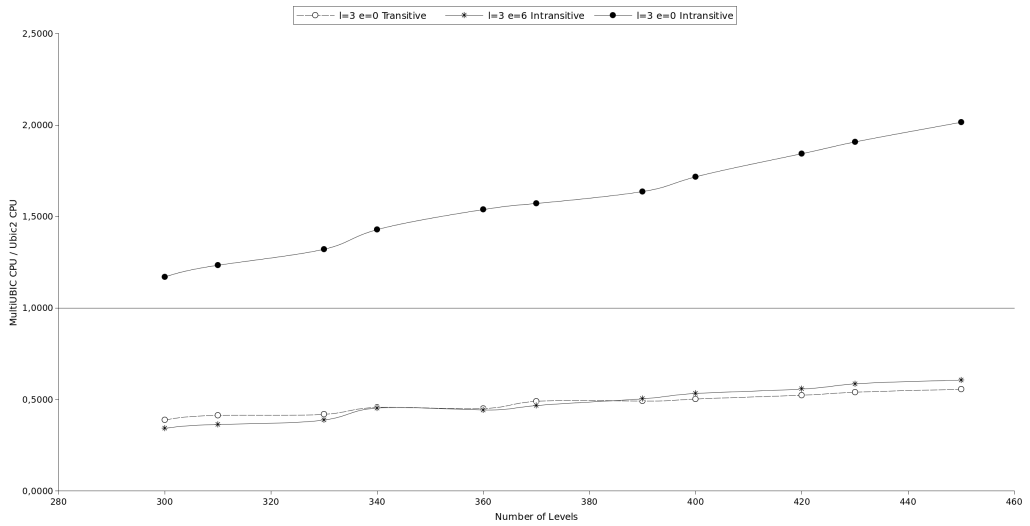


Figure B.17: Experimental comparison of MultiUBIC and UBIC: MultiUBIC become less efficient because of the overhead caused by algorithmic markings.

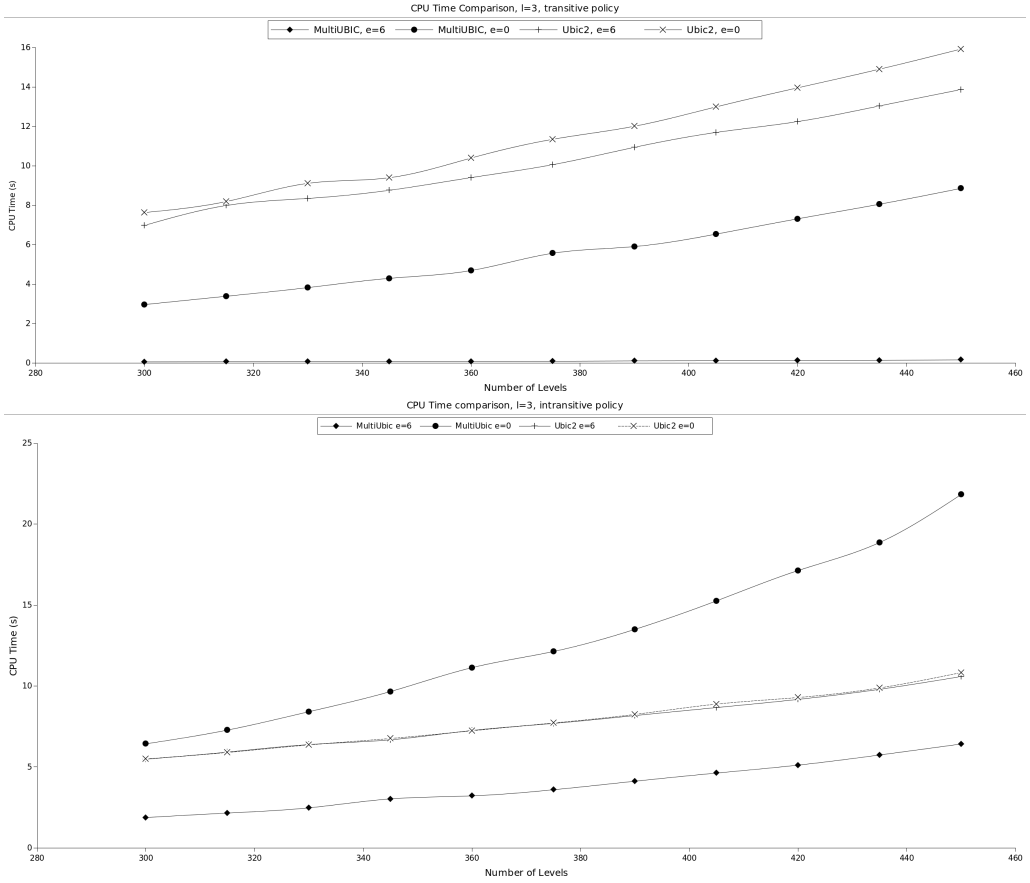


Figure B.18: Experimental comparison: investigating the overhead of intransitive policies.

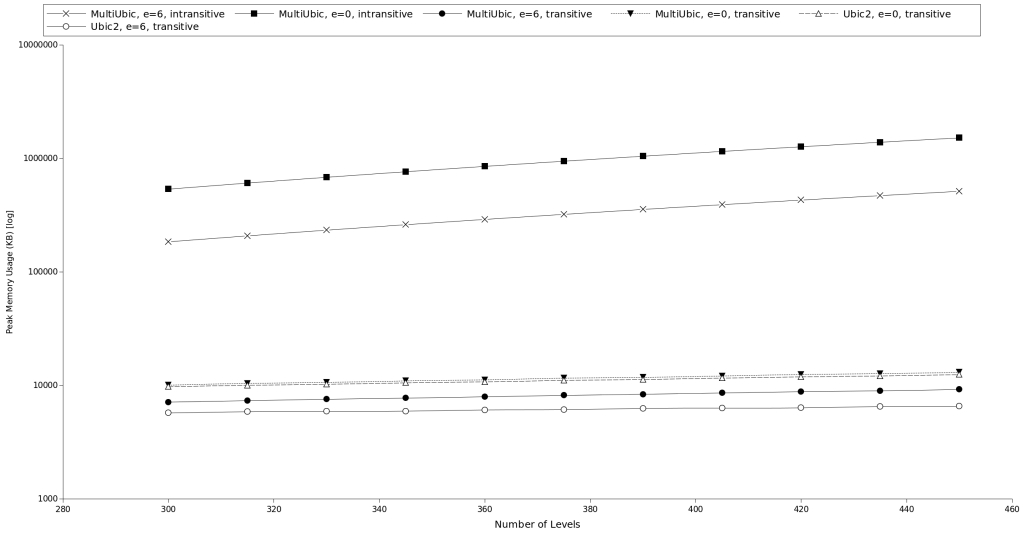


Figure B.19: Experimental comparison: peak memory usage.

a lower peak memory usage, because the prefix generated by MultiUBIC can never be smaller than one of those generated by UBIC. As a matter of fact it could well be exponentially bigger than that of a single two-level problem. This could suggest that, given a sufficient amount of memory, one could optimally solve the causal non-interference by running several UBIC instances in parallel over two-level problems. In principle, this is true, but it is worth observing that the number of instances to be solved in parallel is so high that the amount of memory needed is hardly available on standard front-end machines. Notice how the memory overhead of MultiUBIC seems to be only related to the algorithmic marking of BINI, while for BNDC the memory consumption is aligned with that of UBIC.